Call-A.P.P.L.E.™

Volume V, Number 7 July 1982 $2.50 ($3.00 Canada, £1.75 UK.)

A IS FOR APPLE:
Executive Text Files
Page 9

A CALL —A.P.P.L.E. TUTORIAL:
Examining Directory Sectors
Page 15

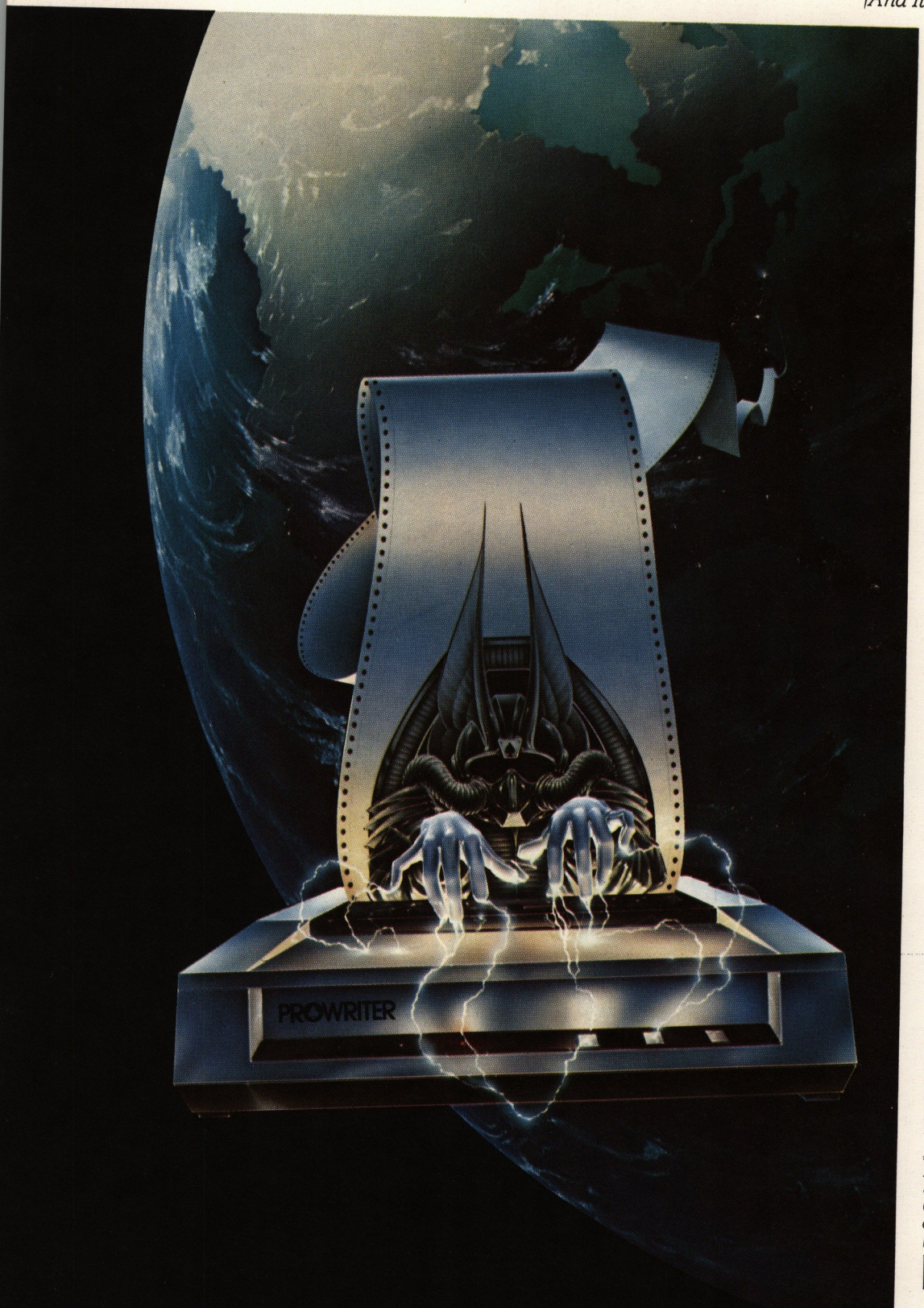
EXPLORING THE WORLD OF A.P.P.L.E.:
Ins and Outs
Page 19

A CALL —A.P.P.L.E. REVIEW:
Word Processors
Page 29

Apple PugetSound Program Library Exchange

THE PROWRITER COMETH.

(And It Cometh On Like Gangbusters.)



Evolution.

It's inevitable. An eternal verity.

Just when you think you've got it knocked, and you're resting on your laurels, somebody comes along and makes a dinosaur out of you.

Witness what happened to the Centronics printer when the Epson MX-80 came along in 1981.

And now, witness what's happening to the MX-80 as the ProWriter cometh to be the foremost printer of the decade.

SPEED

MX-80: 80 cps, for 46 full lines per minute throughput.

PROWRITER: 120 cps, for 63 full lines per minute throughput.

GRAPHICS

MX-80: Block graphics standard, fine for things like bar graphs.

PROWRITER: High-resolution graphics features, fine for bar graphs, smooth curves, thin lines, intricate details, etc.

PRINTING

MX-80: Dot matrix business quality.

PROWRITER: Dot matrix correspondence quality, with incremental printing capability standard.

FEED

MX-80: Tractor feed standard; optional friction-feed kit for about $75 extra.

PROWRITER: Both tractor and friction feed standard.

INTERFACE

MX-80: Parallel interface standard; optional serial interface for about $75 extra.

PROWRITER: Parallel and serial interface standard.

PRICE

Heh, heh.

Distributed Exclusively by Leading Edge Products, Inc., 225 Turnpike Street, Canton, Massachusetts 02021. Call: toll-free 1-800-343-6833; or in Massachusetts call collect (617) 828-8150. Telex 951-624.

LEADING EDGE.™

Boldly Go Where No Drive Has Taken You Before...

Give your APPLE II® a Fourth Dimension — the totally compatible 5¼" drive that takes your system farther, faster. With read/write electronics so advanced that reading errors are virtually eliminated. With a track zero microswitch that keeps boot and track access smooth and quiet. With the ability to read half-track software and up to 143,360 bytes on DOS 3.3®. With similar performance on DOS 3.2.1®, Pascal® or CP/M® operating systems. And, the disk enclosure mates perfectly with APPLE cabinetry.

EXTENDED WARRANTY — We offer a 12 month parts and labor warranty extension at no cost to you.

IMMEDIATE DELIVERY — Immediately available in quantity. We offer same-day shipment if your order is received by noon (PST).

$30 FACTORY REBATE — Receive a $30 factory rebate if the warranty card is received by us on or before August 31, 1982. See participating dealer, or call direct. Suggested Retail Price: $419.00.

TAKE YOUR APPLE TO THE LIMIT WITH A FOURTH DIMENSION DRIVE.
Ask for additional information. Dealer inquiries invited. Contact:

Fourth Dimension Systems
3100 W. Warner Avenue, #7/Dept. A
Santa Ana, CA 92704
(714) 850-1228

FOURTH DIMENSION SYSTEMS

Mastercard and Visa accepted.

APPLE , APPLE II , DOS 3.2.1 and DOS 3.3 are registered trademarks of APPLE Computer, Inc.; Pascal is a trademark of UCSD; CP/M is a registered trademark of Digital Research, Inc.

APPLE PUGETSOUND PROGRAM LIBRARY EXCHANGE:

CLUB INFORMATION

GENERAL OFFICES
304 Main Ave. S., Suite 300
Renton, WA 98055 — (206) 271-4514
Summer Hours: 8-6 Monday — Friday
Thursdays till 7 p.m.
Member Services/Dealer Orders

EDITORIAL DEPARTMENT
6708 39th Avenue S.W.
Seattle, WA 98136—(206)932-8288
Hours: 8-4 Monday - Friday

ADVERTISING REPRESENTATIVES
Grawin Publications
1020 Lloyd Building
Seattle, WA 98101 (206) 223-0861

MEETING NOTICES

7:30 PM, Tuesday, July 20th, 1982

Omega/Bellevue
835 106th N.E., Bellevue

Guest Speaker
Wayne Erickson/DB Management

Special Meeting
7:30 PM, Tuesday, Aug. 3, 1982

Guest Speaker
Mike Markkula/President Apple Computer, Inc.
Contact office for reservations

7:30 PM, Tuesday, August 17, 1982

Computer Empire
616 S.W. 152nd St., Seattle

Special Interest Groups

BULLETIN BOARD

A.P.P.L.E. CRATE
(206) 935-9119
(4 pm - 9 am Pacific Time)

CALL —A.P.P.L.E. HOT LINE

(206) 932-6588
Hours: 9 AM - 3 PM and
6 PM - 10 PM 7 days per week
Pacific Time

See page 85 of the June '82 issue for information on use of the Hotline.

Please also check our list of consultants below for a specialist in your field.

Membership Information

Please direct telephone questions on membership and orders to (206) 271-4514.

One Time Apple—Cation Fee	$25.00
Annual Dues	$20.00
Total	$45.00

Additional postage required for foreign memberships and shipments.

Membership is on an annual, 12 month basis, commencing with the month of enrollment. Upon receipt of total membership fees, new members will receive an Apple—Cation blank, order form and description sheet of club products.

Please allow up to 30 days for a turn around on membership and orders.

Call —A.P.P.L.E. is mailed free twelve times per year as benefit of membership. Subscriptions are not available.

1982 back issues available at $2.50 each.

Call —A.P.P.L.E. bound volumes for previous years are available as indicated on the member order form on the last page.

Institutional memberships are now available. Contact General Office for details.

Prospective authors please send self-addressed stamped envelope (37¢ postage) to Editorial Dept. for a copy of Author's guidelines.

Cover and interior design by Vic Warren Design.

Officers and Directors

Dick Hubert, President	(206) 271-6939
Bob Huelsdonk, Vice President	(206) 362-4910
Dave Lingwood, Secretary	(206) 244-9360
Bob Clardy, Treasurer	(206) 641-1917
Val J. Golding, Director	(206) 932-6588
Ken Smith, Director	(206) 322-4199
Fred Merchant, Director	(415) 921-7792

Office Staff

Dick Hubert	Executive Director
Dave O'Leary	Administrative Assistant
Sally Davies	Admin. of Information Systems
Arliss Sword	Dealer Orders
Patti Hoffmann	Special Projects Coordinator
Gunter Hirt	Bookkeeping, Member Records
Joff Morgan	Technical Support
Cheryle McCalmant	Order Processing
Kris Mortenson	Order Processing
Gayle Rank	Secretary/Receptionist
Mike Christensen	Systems Development
Gina Mocko	Shipping/Receiving

Magazine Staff

Val J. Golding	Editor	(206) 932-8288
Kathryn Hallgrimson	Office Administrator	(206) 932-8288
Ron DeGroat	Pascal Editor	(303) 442-4794
Kevin O. Donohoe	Production Editor	(206) 223-0861
Paul Mosher, M.D.	Staff Writer	(518) 765-4118
Lee Reynolds	Staff Writer	(305) 722-2639
Charles Rusch	Staff Writer	(503) 484-4618
Ralph H. Swerdlow, M.D.	Staff Writer	(916) 920-8709
Grawin Publications	Production, Advertising	(206) 223-0861
David B. Garson	Program Editor	(714) 452-7524

Consultants

Bob Huelsdonk	Apple ///	(206) 362-4910
Dennis Baumgarte	Assembly (Eastern U.S.)	(914) 473-3210
Don Williams	Assembly (Western U.S.)	(206) 242-6807
Michael Thyng	Business	(206) 524-2744
Ted Perry	C.A.I.	(916) 485-1690
William P. Steinberg	DOS (Eastern U.S.)	(212) 921-0663
Art Schumer	DOS (Northern U.S.)	(701) 696-2574
Lee Meador	DOS (Southern U.S.)	(817) 469-6019
Kurt Thams	DOS/Gen'l (Western U.S.)	(714) 870-9316
WM. Reynolds III	DOS/Tech (Western U.S.)	(503) 472-4028
Murray Arnow	Engineering Applications	(312) 291-9210
Cornelis Bongers	European Consultant	010-525511 Ext. 3275
Alan Robinson	Fortran	(503) 757-8234
Tom Deal, M.D.	Handicapped Applications	(206) 634-2847
Randy Halvorson	Hardware Interfacing	(206) 783-9040
Ed So	Hi-Res	(408) 267-3448
Stowe Keller	Mathematics	(804) 293-9441
Fred Merchant, M.D.	Medical Group	(415) 921-7792
Stephen F. Lee	Music	(704) 289-4102
Tom Woteki	Pascal (Eastern U.S.)	(202) 547-0984
Carl Winter	Pascal (Western U.S.)	(206) 454-2720
Randy Warrens	Pilot Language	(608) 837-9566
Hersch Pilloff	Stock Market	(301) 292-3100
Mark Robbins	Telecommunications	(303) 755-6440
Clifton Howard, M.D.	Utilities	(201) 768-9274
Jim Hockenhull	Video Interfacing	(509) 334-4137
Jerry Brieger	VisiCalc	(206) 881-1241
Jeff Finn	Word Processing	(206) 325-0430
Dick Curtiss	Z-80 Card, CP/M	(206) 784-8018

Call-A.P.P.L.E.™

Volume 5, Number 7 July 1982/Menu Val J. Golding, Editor

ENTREES

9 A IS FOR APPLE:
Executive Text Files
Ralph H. Swerdlow/Call —A.P.P.L.E. Staff Writer

15 A CALL —A.P.P.L.E. TUTORIAL:
Examining Directory Sectors
David Huffman

19 EXPLORING THE WORLD OF APPLE:
Ins and Outs: Part 2
Tom Little

29 A CALL —A.P.P.L.E. REVIEW:
Word Processors
Jeffrey K. Finn

A LA CARTE

41 /// CORNERS
Keypad Joystick
Richard V. Grant

43 A CALL —A.P.P.L.E. UTILITY:
Restore To A Line Number Routine
Lee Reynolds/Call —A.P.P.L.E. Staff Writer

45 SO WHAT DID YOU EXPECT?
Unraveling An Applesoft Puzzle
Cornelis Bongers

49 **Patching DOS the Easy Way**
Leighton Paul

51 CALL —A.P.P.L.E. TECHNIQUE:
Imbedding Machine Code In Applesoft Program Files
William J. Cheeseman

59 CALL —A.P.P.L.E. TECHNIQUE:
Dynamic Loading and Linking
John L. Shanks

65 **Dual Drive Master Create**
David L. Deever

67 APPLEMASH:
In aMIDSt Strings
Michael Thyng

75 **DOS Patch for 80 Column Cards**
Glen Bredon

77 A CALL —A.P.P.L.E. UTILITY:
Binary To Text Converter
Val J. Golding with Dave Lingwood

79 **Moving Big Mac Symbol Table**
Thomas G. Williams

HORS d'OEUVRES

3 A.P.P.L.E. Club Information

5 The Editor Bytes Back

5 BOOT
Craig MacIntosh

7 Bytes From A.P.P.L.E

65 So What Did You Expect Dept.?

69 Tomorrow's Apples Today

81 A.P.P.L.E. Meeting Minutes
Michael Thyng/guest secretary

83 WRITE —A.P.P.L.E.
The Readers

87 Advertiser's Index

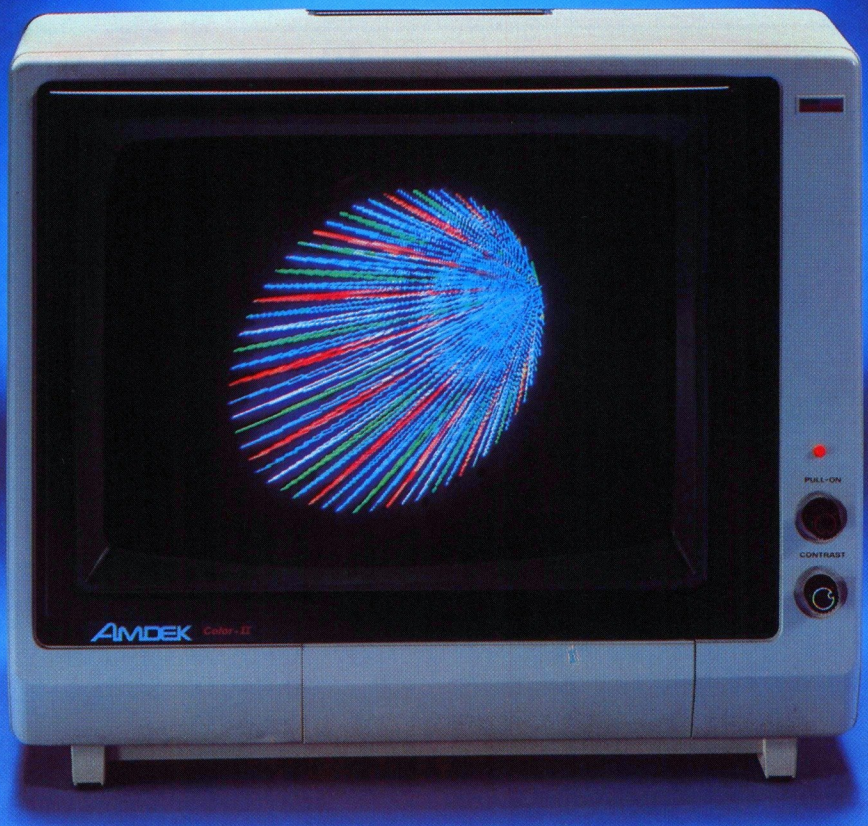
87 A.P.P.L.E. Order Form

88 Call —A.P.P.L.E. Unclassifieds

Apple user groups may reprint without prior permission any portion of the contents herein, provided author, title, and publication credits are given and a written notice of such use furnished to the editorial office. For commercial applications of programs herein, the author should be contacted c/o General Office. Call —A.P.P.L.E. is published monthly by Apple Puget Sound Program Library Exchange, 304 Main Ave. S., Renton, WA 98055. Subscriptions are not available but copies are mailed free as a benefit of A.P.P.L.E. membership.

Entire contents copyright© 1982 by
Apple PugetSound Program Library Exchange

The High-Resolution Color Monitor for Apple II



Amdek's new Color-II monitor with DVM interface board

No video monitor has ever combined Apple II compatibility with exceptional performance like the new 13" Color-II monitor from Amdek.

Color-II features our optional new DVM peripheral board for easy interfacing with your Apple II. And look at the other top-of-the-line features Color-II offers:

- RGB, TTL input for high resolution graphics.
- 80 x 24 character display capability (with expansion card by Vydec).
- 560 (H) x 260 (V) resolution.
- Molded-in carrying handle for easier portability.
- Front-mounted controls for faster fine-tuning.

So if you want to get the most from your Apple II system, get the "most" monitor. Ask your dealer about Amdek's new Color-II monitor with the optional DVM interface board.

What is DVM? The DVM, or "Digital Video Multiplexor," is a low cost interface that allows the Apple II computer to be used with an RGB monitor, such as our Color-II.

Amdek's DVM is software programmable to allow transparent operation, and is parallel with existing Apple text and graphic modes. Three of its channels are used to multiplex the existing Apple text, low resolution and high resolution graphics. The 4th channel allows the use of an 80 character line video board.

Our DVM is also color channel software programmable, enabling you to turn the three color channels on or off by software control. For example, the red and blue channel can be turned off when the 80 character channel is turned on, resulting in a green phosphor video presentation.

For convenient operation, the DVM board may occupy any slot in the Apple II. The DVM also features low-power consumption and low-power schottky logic.

AMDEK

Amdek Corporation, 2420 E. Oakton Street, Suite "E" Arlington Heights, Illinois 60005 (312) 364-1180 TLX: 25-4786

THE EDITOR BYTES BACK

the editor

JULY brings with it thoughts of swimming holes and balmy summer nights; to computer buffs it is an opportunity to head for the [usually freezing] basement and catch up on postponed programming projects. Now is the time to type in **David Deever's Dual Drive Master Create** or **Binary to Text Save** by **Dave Lingwood** and ourselves. Or, if its plain just too hot, you can find your favorite shady spot, pour a tall one, sit back, relax and devour this month's offerings.

Ralph Swerdlow, whose **A IS FOR APPLE** series is absolutely indispensable for novices, provides instructions on how to use **EXECutive Text Files,** unravelling a hitherto unexplored mystery. **Tom Little** continues his exploration of the **World of Apple** with Part 2 of **Ins and Outs,** where he delves into the intricacies of Apple's I/O operations.

For a tutorial, let us recommend **Dave Huffman's Examining Directory Sectors,** if learning more about DOS is your bag. On the other hand, if your interest leans more to applications and word processing, take a look at our continuing series of word processor reviews, hosted by **Jeff Finn.** This time around around, reviewers **Traci Wynn Collins** and **L. K. Bumgarner** look at **Executive Secretary** and **The Correspondent,** respectively.

In brief, those are our cover stories, but far from our total offering. Our European consultant **Cornelis Bongers** returns with the solution to his **So What did you Expect** from last month, in **Unraveling an Applesoft Puzzle;** as usual, his explanation is impeccable. Furthermore, for those readers interested in a preview of Apple's **Super][,** Bongers has submitted an in depth review of the German **Basis 108,** which will appear in the August **Call -A.P.P.L.E.**

If you've gotten tired of swapping diskettes in the drive everytime you run **Master Create, David Deever** has the solution with his **Dual Drive Master Create.** For those of you whose machines sprout wings whenever an 80 column card encounters an inverse catalog, **Glen Bredon** has a patch to DOS to solve the problem in **DOS Patch for 80 Column Cards.** This applies not only to copies of **Big Mac.LC** that were shipped out, but to any 80 column card having problems with inverse characters.

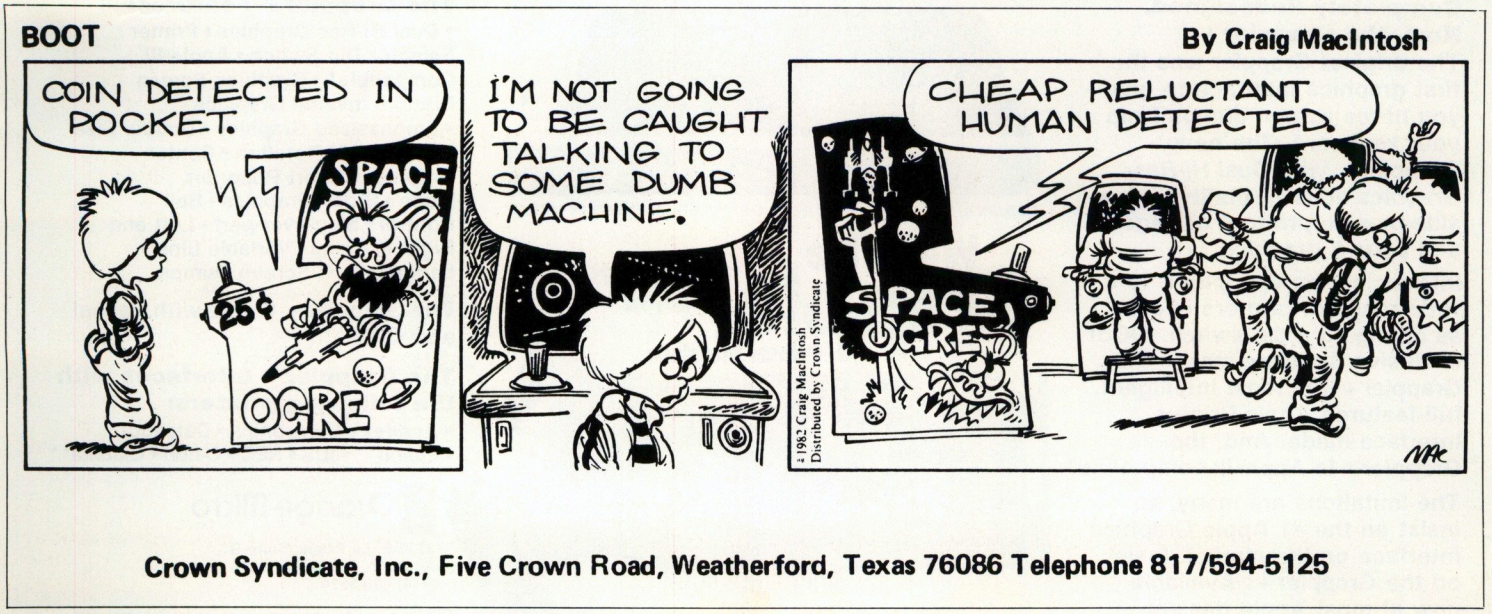
Last month we gave you a sneak preview of our new feature: **Tomorrow's Apples Today.** This time we have a number of interesting items, mostly gleaned from our journey to the Boston Applefest. Each month we will try to report on a select group of new and unusual products. Members should note, however, that publication of these items does not imply our endorsement. It is suggested when a member finds an item to fit their needs, that they investigate thoroughly, both this and competitive products, through their dealer and/or product distributor.

That's only half the story. The whole story can be found by flipping back a page to see what else we have cooked up for you on the menu.

We would like to remind our readers again that they can get a copy of our **Author's Guidelines** by sending a self-addressed envelope with .37 postage to the editorial office. Our page rate remains $50, and we pay a $15 minimum for material submitted for our **Call -A.P.P.L.E. Quickies** and our **So What did you Expect** features.

As a last word, don't pass by those "So What's," there is a lesson in every one. See you next issue.

INTRODUCING



IMAGINE IT...

NEW

NOW WITH DUAL HI-RES GRAPHICS

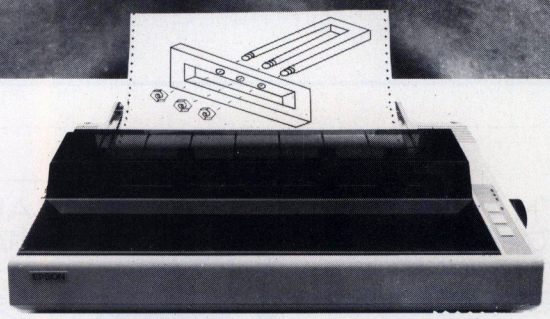
CAPTURE IT.

Completely Redesigned. Now, the Grappler+.
The original Grappler was the first graphics interface to give you hi-res screen dumps from your keyboard. The new Grappler+ with *Dual Hi-Res Graphics* adds flexibility with a side-by-side printout of page 1 and page 2 graphics.

Interfacing the Grappler+ to a wide range of printers is easy as changing a dip switch. 4K of exclusive firmware makes the Grappler+ the most intelligent, full-featured Apple Printer Interface made. And, the Grappler+ is Apple III compatible.*

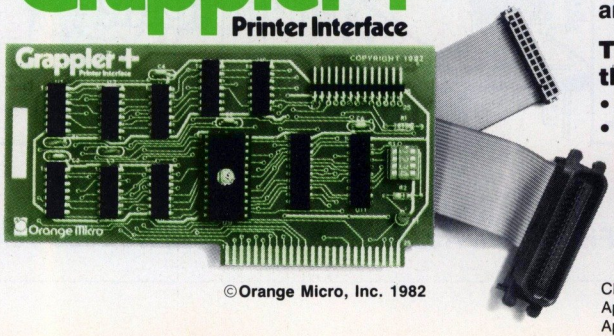
The imitations are many, so insist on the #1 Apple Graphics Interface on the market. Insist on the Grappler+. Available now at most Apple dealers.

*Requires additional software driver.
**Requires graphics upgrade.



ACTUAL APPLE II PRINTOUT USING GRAPPLER™ AND EPSON MX100

With The Grappler+™ Printer Interface



©Orange Micro, Inc. 1982

The Grappler+ Features:
• Dual Hi-Res Graphics • Printer Selector Dip Switch • Apple III Compatible* • Graphics Screen Dump • Inverse Graphics • Emphasized Graphics • Double Size Picture • 90° Rotation • Center Graphics • Chart Recorder Mode • Block Graphics • Bell Control • Skip-over-perf • Left and Right Margins • Variable Line Length • Text Screen Dumps.

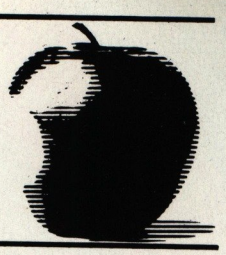
The Grappler+ works with Pascal and CPM.

The Grappler+ interfaces with the following printers:
• Anadex • Centronics • Datasouth • Epson** • IDS • NEC • C-Itoh • Okidata**

Orange Micro Inc.
3150 E. La Palma, Suite G
Anaheim, California 92806
(714) 630-3620

CPM is a registered trademark of Digital Research, Inc. Apple II and Apple III are registered trade names of Apple Computer Inc.

BYTES FROM A.P.P.L.E.



Val J. Golding

GLOBAL Program Line Editor is our big story for this issue. GPLE debuted at the Boston Applefest and, so we have been told, was the "hit of the show." We had a sign up at our booth which read: "You've seen the rest, now get the best." No exaggeration, this, no wild promo. Simply stated, GPLE is without peer.

We have pretty much kept the original PLE in memory whenever our system is up; we would never want to be without it. Now, with the addition of global line edit, search and replace, plus typeahead, 80 column compatibility and 1100 bytes of escape functions definable from the keyboard, GPLE too will become our pride and joy. Member price for Neil Konzen's boon to editing is $38.50.

Next item is one we have been deluged with inquiries on, after members read "Keep your Cool" in the May issue, "...where can I get a fan"? We have the answer to that one in the **Ap.Coolkit.** Specially priced for our members at just $22.50, Ap.Coolkit contains a slow speed Torin fan, a line cord and mounting tape. Give this some serious consideration if you are running into heating problems.

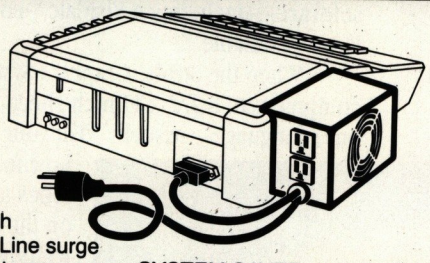
If your subscription to *Softalk* is about to run out, watch for our forthcoming mailer, which will announce a very, very special renewal rate for A.P.P.L.E. members.

Lastly, we are pleased to announce the coming availability to our members of **A.P.P.L.E. Source,** a special closed system of the Source Telecommunications, for access by members only. There will be special features designed strictly for A.P.P.L.E. members, and members who already belong to the Source only need contact us to gain access to the closed system. Best of all, members who sign up now for the Source will receive a 25 percent discount from the regular Source initiation fee. Watch this spot next month to find the effective date of this new service.

Make your Apple II® shine with all new SYSTEM SAVER™

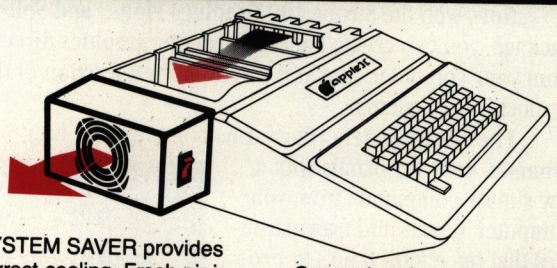
- LINE SURGE SUPPRESSION
- QUIET COOLING FAN
- DUAL AUXILIARY OUTLETS
- FRONT POWER SWITCH

COLOR MATCHED. SIMPLY HANGS ON APPLE.



Power input through SYSTEM SAVER. Line surge suppression protects programs and data by smoothing out dangerous power surges and spikes.

SYSTEM SAVER contains two switched power outlets. You can control your printer and monitor from the front mounted switch.



SYSTEM SAVER provides correct cooling. Fresh air is drawn across the mother board, over the power supply and out the ventilation slots.

Convenient front mounted heavy-duty power switch with pilot light. Switch controls SYSTEM SAVER, Apple II, and both rear outlets.

$89.95 at your local dealer or order direct by phone or mail from:

Kensington Microware Ltd.
300 East 54 Street
Suite 3L
New York, N Y 10022
(212) 486-2802

When ordering by mail include payment of $89.95 plus $2.50 for handling. New York State residents add 5¼% sales tax. By phone payment can be charged to VISA or MASTERCARD.

Dealer inquiries invited.

How to chart your company's fortune without spending one

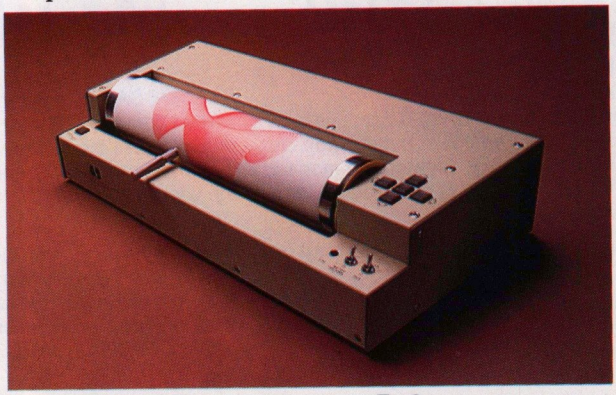
It's a fact. A single chart or graph can tell you instantly what it takes hours to interpret from printouts or other raw data.

Now, with the Strobe 100 Graphics Plotter and Software package, you can create superb hardcopy graphics directly from your computer. And you can do it for a fraction of the cost of most other systems.

The Strobe System transforms complex data into dynamic, colorful visuals with a few simple commands from your computer. Charts and transparencies that once took hours to produce are plotted within minutes. Information can be presented as bar charts, pie charts, curves or isometrics in a variety of colors. And with a resolution of 500 points per inch, the Strobe 100 matches or surpasses the quality of plotters costing thousands of dollars more.

You can also save and modify your graphics through Strobe's menu-driven programs. A broad selection of software—including a VisiCalc*-compatible program—is now available.

When the Strobe Graphics System is interfaced to your computer, an 8-1/2 x 11 inch sheet of paper can speak anyone's language—visually. Visit your local dealer and learn how to start charting your fortunes today. Because a perspective on the present can also be your window on the future.



STROBE INCORPORATED

Strobe Inc.
897-5A Independence Avenue
Mountain View, CA 94043
Telephone 415/969-5130

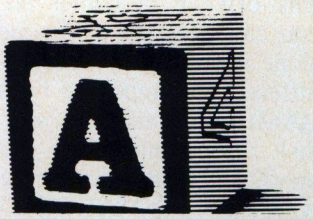
The Strobe Graphics System

Seeing is believing

*VisiCalc is a trademark of VisiCorp.

A IS FOR APPLE:

Executive Text Files



Ralph H Swerdlow/Call —A.P.P.L.E. Staff Writer

TO best understand and most effectively use EXECUTIVE TEXT FILES, the APPLE programmer must fix five facts firmly in mind. These are:

1. Text files consist of nothing more than a series of bytes, one following another, separated by returns, into fields. Each byte can represent either data (arithmetic and/or alphabetic) or instructions. Executive text files contain only instructions that are identical to commands in BASIC.
2. There are two types of BASIC commands; those that require a line number in a program (deferred) and those that do not (immediate execution). Some commands, such as "HOME" or "CALL-936," may be used in either the deferred or immediate modes.

...deferred commands require a line number; immediate do not.

3. All text files are created by writing a BASIC program using the commands of "OPEN, WRITE and CLOSE," with a NAME of the text file that is to be created. The BASIC program used to create an Executive Text file consists of a number of "PRINT" statements that become the commands desired when when that Text file is to be EXECed.
4. When the BASIC program is first run, a Text file is created on the disk, with the name that was selected in the BASIC program. If a catalog of the disk is now made, there will be such a name present, with a "T" preceding it, indicating that it is a Text file.
5. At the command of "EXEC" with the name of the file that was created, the Apple will "PRINT" all of the commands that were in the original BASIC program, as if they were being "typed at the keyboard by a live person"!! If these commands consist of a program, they now can be treated as any other normal program, and run, saved or listed!! If the commands consisted of immediate execution types, they will be carried out at once.

EXECing a file is like typing from the keyboard

With the ground work out of the way, let's start writing some EXEC Text files, and see how they perform. You will need to have a disk booted, as the Text files are saved to disk for later EXECing. We will call the Text files we are to create by the name of "PRACTICE". If you wish to save the files we create, use the name "PRACTICE.1, PRACTICE.2" and so on to later identify which is which. Here is how to create our first file:

```
10  TEXT:HOME
20  N$ = "PRACTICE"
30  D$ = CHR$(4)
40  O$ = "OPEN"
50  W$ = "WRITE"
60  C$ = "CLOSE"
```

Line 10 first sets the screen to its full width, and then clears it. The next five lines establish String statements that we will use to make our program writing easier by reducing the need to type each expression. Line 20 creates a string, N$, that contains the "NAME" that we will give to the Text file. Any time you wish to create a Text file with a different name, it is very simple to change only the contents of the string N$.

```
70   PRINT D$O$N$
80   PRINT D$;"DELETE";N$
90   PRINT D$O$N$
100  PRINT D$W$N$
1000 PRINT D$C$N$
```

These lines first Open the Text file that has the name in the String "N$," then Delete the same file, so that had there been a Text file of the same name on your disk, it will be erased. Note that we may write our Strings one after another and that semi-colons are not needed; this technique reduces both typing time, and the risk of typing errors.

...semi-colons are optional in PRINT statements...

Then they open the file again, and write to the file; line 1000 ends the program by closing the Text file that was created. We will leave sufficient space to add more lines, thus the last line is numbered 1000.

Save this to a disk, using the command "SAVE TO MAKE PRACTICE." This BASIC program will serve as the skeleton upon which you will hang the flesh to make Exec Text files in the future!! A catalog of the disk now shows a BASIC program called, "TO MAKE PRACTICE."

Now, add line 110:

```
110 PRINT "HOME"
```

EXECUTIVE TEXT FILES

List the program, and then run it by typing, "RUN." Note that the disk drive light comes on and that you can hear the drive motor. This alone should tell you that something different is happening because most BASIC programs do not start the disk drive when they are run. Do another catalog, and now there is both the "TO MAKE PRACTICE" and "PRACTICE" on the disk, with a "T" preceding the latter. List the program once more. It is still there!! Now type:

```
"EXEC PRACTICE"
```

If all went as expected, your screen cleared, as if *you* typed "HOME." Here is what happened. When the command "EXEC PRACTICE" was EXEC'ed, the Apple printed out the contents of the Text file, which at this point consisted only of the command "HOME", and the computer did what it was told to do, cleared the screen.

Now type "LIST," and there is your program once more, demonstrating that EXEC'ing a Text file does not erase a program in memory. Try it again, list several times, or fill the screen with garbage of some sort. Type, "EXEC PRACTICE," and the screen clears. This is a very important characteristic of Text files; they do not destroy programs nor does listing a BASIC program or in fact, the running of a BASIC program, impair the later EXEC'ing of a Text file!!!

...EXEC files do not destroy programs...

Now lets add more flesh to our original program, by adding line 120. Load the program "TO MAKE PRACTICE," and add line 120:

```
120  PRINT "NEW"
```

Now run this program. The disk drive again comes to life, and a new Text file, though with the same name of "PRACTICE" is created, but this new Text file contains an additional command, line 120.

List several times, and fill the screen with words. Type, "EXEC PRACTICE," and as expected, the screen clears. So far not much difference from the first program we EXEC'ed. However if you try listing the program, it is gone!! The command in line caused the trusty Apple to type, "NEW," and our computer did so!!

Now add line 130 and 140 to the program, "TO MAKE PRACTICE."

```
130  A$ = "ISN'T THIS FUN."
140  PRINT "PRINT A$"
```

Save this program, (by typing, "SAVE TO MAKE PRACTICE") and then run it, creating a new Text file, but one with the same name of "PRACTICE." Now when it is EXEC'ed, the Apple will print the contents of the String "A$".

You can demonstrate the same thing by just typing:

```
"A$ = "ISN'T THIS FUN."
"PRINT A$"
```

because these are immediate commands that will be executed by merely typing each, followed by a RETURN of course, at the keyboard. Your Apple just did the work for you when the file was EXEC'ed!!

Many lines in a BASIC program look like this:

```
PRINT "ISN'T THIS FUN"
```

The easiest way to print such lines is to make a String which contains the expression, as we did earlier. Here is another method of doing the same thing. Add lines:

```
150  Q$ = CHR$(34)
160  PRINT "PRINT";Q$;"AND
     EDUCATIONAL ALSO"
```

This concantenates (joins together) the statements inside the ["] with the String Q$; CHR$(34) is the quotation mark! After running the previous program, you can prove this by typing, "PRINT CHR$(34)," followed by a RETURN, and a ["] will appear. this is a fine way to put quotation marks into statements that you wish printed.

The commands "HOME," "NEW," "PRINT" and many others may be used in the immediate or deferred mode, thus they do not require a line number to function. When they are "typed" as the Text file is EXEC'ed, these commands are EXECuted at once by the computer. This might be a good time for the reader to review BASIC to remember which instructions are used only in the deferred mode, which only in the immediate, and which in both.

...commands are executed in the order they were entered...

The commands are EXECuted in the order that they were first entered into the BASIC program that was used to create the Text file, but if they have line numbers, our computers arrange the commands into proper sequence.

Now change the contents of the String A$ (in line 130) to anything else and rerun the program. The contents of your new String will be printed.

What would happen in you now change line 130 of the last program to:

```
130  ISN'T THIS FUN
```

and delete line 140?

Note the lack of quotation marks and omission of the world "PRINT." Go ahead and try it; first run the program and then EXEC it. You will get the message, "?SYNTAX ERROR." Do you know why? You can do the same thing from the keyboard of your Apple. Clear the screen, and type, ISN'T THIS FUN, (without quotations) and a return; the same error message is printed, because this statement is not recognized by the computer as a legitimate BASIC command!! So when the same statement is part of the data contained within a Text file, and then that file is EXECuted, it is exactly as if you typed the same sentence at the keyboard!! The lesson to be learned by this example is that the data statements in any Executive text file *must contain only legal commands that the Apple can execute.*

How then to enter deferred mode commands into the "MAKE PRACTICE" program? Here is how:

First load the program, then list it. You should have lines 10–100, and the line 1000. If not, then delete all other lines, and save MAKE PRACTICE so that you will have it on disk for later use.

Now add these lines:

```
110 PRINT "10 HOME"
120 PRINT "20 FOR I = 1 TO 20"
130 PRINT "30 PRINT I"
140 PRINT "40 NEXT I"
150 PRINT "50 END"
160 PRINT "DEL 1,9"
170 PRINT "DEL 51,2000"
```

Now run this program and a new Text file will be created, still with the name of, "PRACTICE." Can you see what will happen when you EXEC PRACTICE? Lines 110 to 150 will be printed to the computer. When you first EXEC the Text file, nothing will appear on the screen except for a series of Applesoft prompts, and nothing else. At this point, you may list the program, and there is the BASIC program typed in lines 110-150 in our last MAKE PRACTICE. Now the new BASIC program can be run and it will print the digits as expected. You may merely EXEC the Text File, then type, "RUN," and the Apple will do so. To see what the computer is doing, type the command, "MON C,I,O," and you will be able to see each line being "typed" by the computer as the Text file is EXECed....

First, line 110 is EXEC'ed printing, "10 HOME." Next lines 120 to 150 (which continue the printing of our new BASIC program) are typed to the screen. These lines of the Text file create the lines of the BASIC program with line numbers; next lines 160 and 170 delete all that created the Text file leaving only the BASIC program in memory. The command "DEL" can be used in both immediate or deferred modes, so lines 160 and 170 could have printed line numbers to the BASIC program. The entire process may be watched if the command "MON I,C,O" had been in effect. The command, "NOMON C,I,O" makes the the EXECing of a Text file invisible, except for the Applesoft prompts. This new program may be saved (for example, type: "SAVE PRACTICE.A") and then listed or run whatever desired. It is very important to use a different name for the BASIC program that was created, if it is to be saved to the disk, else there will be a Text file and a BASIC file with the same names; a very confusing situation for both you and your Apple!!

"NOMONCIO" will hide the EXECing of a text file...

For all practical purposes, Text files may be of any length desired, limited only to available memory. The skeleton of MAKE PRACTICE has sufficient space between lines 101 and 999 to make almost any Text file that you may need, limited only by your imagination and requirements. Remember that by changing the contents of the String N$ in line 60, and the data in lines 101–999, you can create as many EXEC Text files as you want.

Why use Text files at all? They are very useful for many programming purposes. For example, working on a long BASIC program that may require frequent changes. The program may be retained as a Text file, and then EXECed as needed. The changed program then can be saved (again with another name) to the disk. This process can be repeated as often as necessary, without the need to retype the program each time. If you have many favorite subroutines, they can be made into Text files and EXECed into any programs under development. Some programmers keep a disk which contains only Text file subroutines that are of importance to them.

EXEC'ing a Text file does not destroy a program in memory as loading or running a program does, so your program is not lost, but is still present for you to use. The commands in the Text file will be added to the program in memory, and become part of your new program; this is very useful in program development!

If there is a potential conflict due to identical line numbers, the RENUMBER utility may be employed to change the line numbers to avoid this problem; PROGRAM LINE EDITOR will also do the job.

The program called "CAPTURE" is a BASIC program that creates a Text file when it is run!! You can find it on page 76 in the Apple DOS Manual. CAPTURE is added either at the beginning or the end of a BASIC program in memory. The Text file that it creates will have the name, "LISTING" which will be found in the catalog after the program is run. The data statements of this Text file will consist of the lines in the BASIC program that was in memory when CAPTURE was run! You can then use the RENAME command to change LISTING'S name to something more appropriate to what the Text file does; remember to use a different name than that of the original BASIC program. I prefer to add the prefix "TEXT." to the Text file name; this tells me what it is and that it is and that it is a Text file based BASIC program still saved on my disk. I also have renumbered "CAPTURE" to start at line 30000, because most of my programs are much shorter, and thus there are no line number conflicts when CAPTURE is appended to the end of of my program in memory. I have also changed old line 6 of the Apple Manual to read "LIST 1,29999." I have created a Text file called, "MAKE CAPTURE" which is kept on a utility disk. It is very easy then to EXEC "MAKE CAPTURE", which creates a BASIC program and append it to any BASIC program that I am working on. Here is how to do the same.

...add the prefix "TEXT." to a text file.

```
30000 REM TO MAKE CAPTURE
30001 D$ = CHR$(4)
30002 PRINT D$;"OPEN
      LISTING"
30003 PRINT D$;"WRITE
      LISTING"
30004 POKE 33,33
30005 LIST 1,29999
30006 PRINT D$;"CLOSE
      LISTING"
30007 TEXT:END
31001 PRINT D$;"OPEN MAKE
      CAPTURE"
31002 PRINT D$;"WRITE MAKE
      CAPTURE"
31003 POKE 33,33
31004 LIST 30000,30008
31005 PRINT D$;"CLOSE MAKE
      CAPTURE"
31006 TEXT:END
```

EXECUTIVE TEXT FILES

When this program is RUN it will create a Text file called, "MAKE CAPTURE." Lets study this program.

Lines 30001 to 30007 are the lines that constitute the Text file "CAPTURE." Note that each data statement consists of an immediate type of command, and thus does not require a line number. When these lines are EXECed, they create a Text program, "LISTING" that captures a BASIC program in memory.

Lines 31001–31007 capture the earlier lines in a Text file called, MAKE CAPTURE. Thus, we have written a BASIC program that creates a Text file that captures a Text file that creates a BASIC program that captures a BASIC program into a Text file!!!

Save this program, with the name of "TO MAKE CAPTURE." When it is run, it will create a Text file with the name of "MAKE CAPTURE." Now, whenever this Text file is EXECed, the BASIC program called, "CAPTURE" will be appended to any BASIC program in memory, starting at line 30000. When you run this BASIC program, a Text file containing the commands of your BASIC program will be created. As an interesting experiment, once "CAPTURE" has been appended, typing: "RUN 30000" will accomplish the same task. The Text file, "MAKE CAPTURE" can be transferred to each of your working disks, (using F.I.D, for example) and whenever you need the BASIC program of "CAPTURE," all you need to do is EXEC MAKE CAPTURE, and there it is; CAPTURE will be appended to any BASIC program in the Apple's memory at the time!!

Not everyone has a MODEM, but those that do, are able to send Text files by phone to others who also have one. BASIC programs can not easily be sent by phone, but with CAPTURE, an Apple user is able to convert any BASIC program to a Text file, and later transmit the created Text file over the phone. The receiver can then save the Text file, EXEC the file, and have the BASIC program.

In summary:

1. EXECutive Text files are Text files that contain in their data lines legal BASIC commands that may be either deferred or immediate types.
2. Commands that are of the deferred type must have a line number.
3. The command "EXEC" is similar to the command "RUN,"* and when issued causes the contents of the data statements of the Text file to be printed without erasing the program in memory. The BASIC program may then be run, listed or saved as can any BASIC program.
4. Text files are created using the DOS commands of "OPEN, WRITE, AND CLOSE," with Control "D" or CHR$(4) preceding each command. A specific name must be incorporated into each command. The Text file created will have the name that was chosen.
5. If a data statement in a Text file is not a legal BASIC command, an error message will be printed by the computer, saying "?SYNTAX ERROR" whenever the Text file is EXEC'ed.

Text files are not difficult to master, and they do offer many benefits to the Apple user. Text files containing legal BASIC commands add great power to our computers. Go ahead and try them; you not only will have fun, but you are sure to learn something helpful to your computing.

** In fact one could say that RUN will run a deffered mode program and EXEC will run an immediate mode program. ...ed.*

INTRODUCING THE AUTOMANIAC SYSTEM

- It runs your Apple 24 hours a day! • It runs programs and series of commands that you schedule!
- It brings Apple's Exec facilities to the Programmer and Non-Programmer!
- It can control electrical devices in the Apple's environment on a scheduled basis!

Why not run some of those routine programs at night when the computer is free, so that during the day it is free for you!

FOR EXAMPLE:

During the night, on Mondays, Wednesdays, and Fridays, you can schedule the program that sorts your Accounts Receivable data. On the 15th of every month you can schedule a series of programs that will turn your printer on at 7:00 P.M., print out the bills at 7:05 P.M., and after all the printing is finished, the printer is turned off. During the day, your computer won't be bogged down doing file maintenance but instead will be free to access information.

The **AUTOMANIAC SYSTEM** is compatible with any one of the following clocks:

• Applied Engineering Time II* • Mountain Hardware Apple Clock* • Mountain Hardware CPS card* • Superclock II*
• CCS Calendar Clock* • Computime Compuwatch* • Thunderware Thunderclock* • Prometheus Versacard* *Trademarks

The system requires: 32K or 48K Apple II Plus*, 3.3 D.O.S., An 80 col. printer and Mountain Hardware Introl/X-10 card* are optional.

PRICE $100.00 - Order now and become an **AUTOMANIAC!** You'll get a GSW notebook, a 34-page easy-to-use operating manual, and a 3.3 D.O.S. diskette.

Call (515) 280-7916 to order by Mastercard or VISA. Send check or money orders to:

GSW

THE GEEGERY SOFTWARE WORKS
P.O. BOX 8028
DES MOINES, IOWA 50301

A Fast, Easy, Inexpensive Approach to Word Processing

NEW

PIE Writer:
Word Processing System

Hayden is thinking about your future. We know you're concerned with saving time and money . . . with increasing efficiency and productivity . . . and that is the reason for PIE Writer!

Now you can turn your Apple II into a complete word processing system. PIE Writer, a powerful editor and formatter, offers the versatility and machine efficiency of the larger systems—at a fraction of the cost!

Now all your documents and correspondence—reports, memos, direct mail pieces, even "personalized" form letters—can be composed, edited, formatted, and printed quickly and accurately. Simple keystroke commands let you alter characters, words, sentences, even entire paragraphs!

Learning to use PIE Writer is as easy as learning to type. There are many user-oriented features, designed for first-time users as well as serious programmers: an easy-to-understand, indexed instruction manual, a carefully designed reference card, an interactive tutorial with each diskette, a "help" screen listing key command descriptions for reference as text is edited, and an easel binder for at-a-glance aid.

If you're looking for high-quality at a low cost, PIE Writer is your best investment. Order yours today! 12009, Standard; 13409, Smarterm, Sup'R'Term, Videx

Available at your local computer store

or order direct;

1-800-631-0856

in New Jersey 201-843-0550, ext. 382

Take a look at some of PIE Writer's powerful new features:

- Incremental Spacing–adds fractions of space between words for text justification.
- Fast page scrolling.
- Call command–allows access to machine language subroutines.
- Saves formatted output on disk file.
- Works with 40 or 80 column display boards.
- Word Tabbing–cursor stops at beginning of every word, in addition to set stops.
- Tab stops can be saved on file.
- Word Delete–deletes entire words at a time.
- Lower case capabilities.
- Prints **BOLD**.
- Centers.

All this and more . . .
for only $149.95!

Upgrade your PIE 2.0: Send $75.00, check or money order, and original disk (at sender's risk) for upgraded version.

Hayden

Mail To: **Dept. #CA 626 • Hayden Book Company, Inc. 50 Essex Street • Rochelle Park, NJ 07662**

Please accept my order and send me the PIE Writer System for the following display board.

Please Check: ☐ 40 column (#12009) ☐ 80 column (#13409)

☐ My money order for $149.95 is enclosed; or

☐ Please charge my ☐ Visa ☐ MasterCard

Name _____

Address _____ Apt. _____

City _____

State _____ Zip _____

MasterCard/Visa# _____ Expires _____

Signature _____

Residents of NJ and CA must add sales tax.

CA 6/82-006 A501

GET FREE SOFTWARE FOR YOUR COMPUTER!

HOW? JUST ORDER ANY OF THE ITEMS BELOW, AND SELECT YOUR FREE SOFTWARE FROM THE BONUS SOFTWARE SECTION, USING THE FOLLOWING RULE: FOR THE FIRST $100.00 WORTH OF MERCHANDISE ORDERED TAKE 1 ITEM; FOR THE NEXT $200.00 WORTH OF MERCHANDISE ORDERED TAKE ANOTHER ITEM; FOR THE NEXT $300.00 TAKE A THIRD ITEM, ETC. ALL AT NO COST.

HARDWARE by APPLE COMPUTER

APPLE II + , 48K	1199	PASCAL	150
FLOPPY DR. + CNTRLR	545	FLOPPY DRIVE	465
APPLE III, 128K	2999	PILOT	125

We carry the rest of the APPLE line at low, low prices! CALL!

OTHER HARDWARE for APPLE

D.C. HAYES:			
Micromodem II	285	Smartmodem	225
MICROSOFT:			
Z80 Softcard	269	16K Ramcard	139
MOUNTAIN COMPUTER:			
Expansion Chassis	559	Music System	339
A/D + D/A Card	299	CPS Multi-function	169
X/10 Control Card	169	Super Talker	169

CALL FOR MORE PRICES! WE CARRY FULL LINE!

CALIFORNIA COMPUTER SYSTEMS:			
Centronics Par. Int.	115	A/D Converter	105
Async. Serial Int.	135	Calendar/Clock	101

CALL FOR MORE PRICES! WE CARRY FULL LINE!

VIDEX:			
80 Col. Bd. & Softswitch	235	Enhancer II	125
Enhancer I	105	Softswitch	29
MORE OTHER HARDWARE FOR APPLE:			
SSM AIO-II	195	Keybd. Co. Num. Keypad	129
SSM Serial AS10	115	Sunshine Joystick	39
SSM Par. AP10	99	Game Paddles	29
Novation Apple Cat	319	Shadow/Vet	885
Versawriter Tablet	249	SUP'R'MOD	29
Prac. Periph. Microbuff. (32K)	249	Prac. Periph. Microbuff. (16K)	210

OTHER SOFTWARE for APPLE:

PERSONAL SOFTWARE/VISICORP:			
Visicalc 3.3	195	Visifiles	199

CALL FOR MORE PRICES! WE CARRY FULL LINE!

MICROSOFT:			
APPLE Fortran (Z80)	129	APPLE Cobol (Z80)	499
TASC Basic Compiler	139	MBASIC Compiler (Z80)	299
TIME Manager	125	MuMath	199
ALDS	99	M/SORT	149
MICRO-PRO:			
Wordstar	225	Mail-Merge	99
Spellstar	149	Data-Star	199
Super-Sort	149	Calc-Star	149
PEACHTREE: BIZ Packages, all			199
CPA BIZ Packages, all			195

MORE OTHER SOFTWARE for APPLE:

DB Master	179	DB Master for CORVUS	399
Data Factory 5.0	239	PFS	85
ASCII Express	55	Dakin 5 Deprec. Planner	299
Sorcim Super Calc	189	Dakin 5 BIZ Bookkeeper	299
Howard Tax Prep.	115	Broderbund Payroll	325
Howard Real Estate Anal	129	BPI Accounting Pkgs/ea	325
Synergistic "Data Reporter" Manages, Plots & Edits Data!			189

WORD-PROCESSORS & SPELLERS for APPLE:

Wordstar CP/M	225	EZWriter Prof. Sys.	215
Magic Wand CP/M	299	MUSE Super Text 80	150
Executive Secretary	199	Wordpower	50
Letter Perfect	125	Hebrew II	55
Magic Window	85	Screenwriter II	110
Spellguard	219	Spellstar CP/M	149

Word Handler (Gives 80-Col. & Lower-Case with no board!) 199

PRINTERS

EPSON:			
MX80	449	MX80 F/T	549
MX100 w/Graftrax	729	MX70 w/Graftrax	285
APPLE Intfce/Cbl	85	GRAFTRAX	60
GRAPPLER Intfc	149	2K Buffer Serial Card	135
MX80 Ribbon	15	MX100 Ribbon	24
C.ITOH:			
F-10 Daisy Wheel (Par)	1495	F-10 Daisy Wheel (Ser)	1495
Pro-Writer (Par/Ser)	599	Pro-Writer (Par)	499
F-10 Tractor Option	225	Printer Interfaces	CALL
NEC:			
PC-8023A	495	NEC 7710 Daisy	2345
QUME:		**DIABLO:**	
SPRINT 9/45	1995	630 R/O	2099
OKIDATA:			
Microline 82A	495	Microline 80	375
Microline Tractor	59	Okigraph I	79
Microline 83A	799	Microline 84 (Par)	1099
IDS:			
560 with graphics	1095	Prism-Print Software	49
Prism 80 (Basic)	899	Prism 132 (Basic)	1050
Auto Sheet Feed	125	Prism Color	325
Sprint Mode (200 cps)	125	Dot Plot Graphics	85

ADD-ON MEMORY CARDS & DISK DRIVES FOR APPLE

MEMORY:			
Microsoft 16K Ramcard	139	Saturn 32K Card	199
Legend 128K Ramcard	649	Saturn 64K Card	369
Legend 64K Ramcard	299	Saturn 128K Card	525
SVA 256K APL-Cache	1045	Prometheus 128K	439
AXLON 320K Ram Disk	1149	16K of 4116, 200 NS Mem.	25
APPLE-COMPATIBLE FLOPPIES by MICRO-SCI:			
With Controller:		No Controller:	
A35 Exact Replacement	460	A35 Exact Replacement	415
A40 40-Track	489	A40 40-Track	399
A70 70-Track	599	A70 70-Track	499
8" FLOPPY DISK SYSTEMS:			
Vista Dual SSDD	1299	Vista Dual DSDD	1599
SVA AMS8000 Dual SSDD	1945	SVA AMS8000 Dual DSDD	2595
SVA ZVX4 Quad Cntrllr.	495	SVA Disk 2 + 2 Cntrllr.	359
CORVUS HARD DISKS:			
6 MB Hard Disk	2249	Apple Interface	175
11 MB Hard Disk	3945	Other Computer Intfce	CALL
20 MB Hard Disk	4769	Mirror Back-Up	675

MONITORS, PLOTTERS & PERIPHERALS

MONITORS:			
Zenith 12" Green	125	Zenith 13" Color	359
Amdek 12" Green	135	Amdek 13" Color	359
BMC 12" Green	119	BMC 12" Color	349
Electrohome RGB Intfce	275	Electrohome RGB 13" Clr	725
PLOTTERS:			
Watanabe 1-Pen	1150	Watanabe 6-Pen	1400
Strobe Plotter 1-Pen	699	Strobel Apple Intfce	99
Houston Inst. DMP-3	929	Houston Inst. DMP-4	1185
OTHER PERIPHERALS:			
Scott Shadow/VET	885	Street Echo II Synth	189
Votrax Type-N'-Talk	325	Computer Sta. Dithertizer	269
ALF 9-Voice Music Bd.	155	Comp. Sta. Video Camera	375
ALF 3-Voice Music Bd.	199	Comp. Stat. Both above	599

COMPUTER SYSTEMS

ATARI			
ATARI 800 (16K)	629	ATARI 400 (16K)	335
810 Disk Drive	449	825 Printer	599
16K Ram Memory	89	850 Interface	159
Microsoft Basic	69	830 Modem	149
INTEC 32K Ram Memory	129	Telelink Cartridge	25
ATARI 800 (48K)	739	AXLON 128K Ram Disk	550
		ATARI 400 (48K)	435
OSBORNE			
Osborne 1	1695	Printer Cable	55
XEROX			
820-1 System w/5" Dr.	2450	820-2 System w/8" Dr.	2950
CP/M Op. Sys.	159	Wordstar	419
Super Calc	199	DIABLO 630 Printer	2099
Systems-Plus BIZ. SOFTWARE, PER MODULE			CALL!
NEC			

Full Line at Low, Low Prices! CALL!

COMMODORE VIC			
VIC 20 Computer	259	VIC 1515 Printer	335
VIC 1540 Disk Drive	499	VIC 1211A Supr-Expndr	59
VIC 1530 Datasette	69	VIC 1011A RS-232 Port	45

GENERAL CP/M SOFTWARE

MICROSOFT:			
Basic 80	275	Edit 80	139
Basic Compiler	299	Mu Math/Mu Simp	199
Fortran 80	339	Mu Lisp/Mu Star	159
Cobol 80	499	M-Sort	128
Macro 80	139	Z-80 Softcard/Apple	269
MICROPRO:			
Wordstar	275	Calcstar	199
Mailmerge	89	Supersort	165
Spellstar	165	Custom. Notes	275
Datastar	239		
ASHTON-TATE:			
dBASE II	475	dBASE II Guide	29
SORCIM:			
Super Calc	189		
FOX-GELLER:			
Quickscreen	129	dUTIL	69
Quickcode (Writes programs for dBASE II)			199
ISA:			
Spellguard	219	SP/LAW	99
PEACHTREE:			
Gen. Ledger	399/40	Inventory	399/40
Acct. Rec.	399/40	Magicalc	269/25
Acct. Pay.	399/40	Sales Invoicing	399/40
Payroll	399/40		
SYSTEMS PLUS			
All Modules	415/EA		

S-100 BOARDS

HAYES S-100 MODEM	325	HAYES SMARTMODEM	219
HAYES CHRONOGRAPH	189	CCS 64K RAM BOARD	525

FLOPPY DISKS

Elephant 5.25"		Elephant 8"	
Soft SSSD/bx	25	Soft SSSD/6x	29
3M 5" Soft SSSD/bx	27	3M 8" Soft SSSD/bx	32
Maxell 5" Soft SSSD/bx	31	Maxell 8" Soft SSSD/bx	35
DYSAN 5" Soft SSSD/bx	37	DYSAN 8" Sof SSSD/bx	49

BONUS SOFTWARE SECTION!

Let us acquaint you with MESSAGE-MAKING SOFTWARE. Just place the disk in the APPLE, enter the text, and colorful, dynamic messages appear on the screens of TV sets connected to the computer. Use the software to broadcast messages on TV screens in schools, hospitals, factories, store window, exhibit booths, etc. The following program is our latest release:

SUPER MESSAGE: Creates messages in full-page "chunks". Each message allows statements of mixed typestyles, typesizes and colors, in mixed upper and lower case. Styles range from regular APPLE characters, up to double-size, double-width characters with a heavy, bold font. Six colors may be used for each different typestyle. Vertical and horizontal centering are available, and word-wrap is automatic. Users can chain pages together to make multi-page messages. Pages can be advanced manually or automatically. Multi-page messages can be stored to disc or recalled instantly.
REQUIRES 48K & ROM APPLESOFT . . . . . $50.

APPLE PLOTS YOUR DATA & KEEPS YOUR RECORDS TOO! APPLE DATA GRAPH 2.1: Plots up to 3 superimposed curves on the Hi-res Screen both X & Y axes dimensioned. Each curve consists of up to 120 pieces of data. Graphs can be stored to disc and recalled immediately for updating. Up to 100 graphs can be stored on the same disc. Great for Stock-market Charting, Business Management, and Classroom instruction!
REQUIRES 48K & ROM APPLESOFT . . . . . $35.

APPLE RECORD MANAGER: Allows complex files to be brought into memory so that record searches and manipulations are instantaneous. Records within any file can contain up to 20 fields, with user-defined headings. Information can be string or numeric. Users can browse thru files using page-forward, page-backward or random-search commands. Records can easily be searched, altered or sorted at will. Files can be stored on the same drive as the master program, or on another, if a second drive is available. Records of files can be printed, if desired. Additional modules coming are a STATISTICS INTERFACE, CHECKBOOK, MAILING LIST & DATA-ENTRY.
REQUIRES 48K & ROM APPLESOFT . . . . . $40.

APPLE LITERATURE DATABASE: Allows rapid retrieval (via keywords) of references from total APPLE literature thru 1980, on 5.25" disk. Each entry in the data base consists of the article, author-name, periodical-name, date of issue, & page nos. The database is intended to support large magazine files which would require lengthy manual searching to recover information. Annual updates will be available.
REQUIRES 48K & ROM APPLESOFT . . . . . $60.

WORDPOWER: Is a simple, powerful, low cost, line-oriented word-processor program. It offers a fast machine language FIND & REPLACE. Text can be listed to screen or printer, with or without line-numbers. Lower-case adaptors are supported. You can merge files, move groups of lines, and easily add, change, or delete lines. WORDPOWER can be used to create and maintain EXEC files. It can also be used as a rapid, unstructured, information-storage and retrieval system via its rapid search capabilities.
REQUIRES 48K & ROM APPLESOFT . . . . . $50.

LABELMAKER: Allows users to quickly create address labels. A given label may be generated in any quantity from 1 to 32767. Space is allowed on labels for a personal and company name, but the space is automatically closed up if only a personal name is entered. Space is also allowed for foreign countries. The program can also generate labels for price-tags, part numbers and mail-messages such as "RUSH", "FRAGILE", etc. A self-incrementing feature allows theatre-tickets to be produced, with a date, and numbers running from a000 to z999. An editor is provided for editing labels prior to printing. All labels may be saved to disk for instant recall.
REQUIRES 48K & ROM APPLESOFT . . . . . $35.

Above software for APPLE DOS 3.2/3.3 only. Call for BONUSES for other systems.

TO ORDER: Use phone or mail. We accept VISA, MASTERCARD, COD's, personal checks & money orders. Add 4% for credit card. Customer pays handling on COD orders. Foreign orders must be in American Dollars & include 10% for handling. Connecticut residents add 7.5% sales tax. Prices subject to change without notice.

Not responsible for typographical errors. Prices subject to change without notice.

CONN. INFO. SYSTEMS CO.
(203) 579-0472
218 Huntington Road, Bridgeport, CT 06608

A CALL —A.P.P.L.E. TUTORIAL:

Examining Directory Sectors

David Huffman

THE DOS Reference Manual (DOS 3.2 or 3.3) explains the arrangement of the information in a typical sector of the diskette directory on pages 129-131. However, we can usually only examine this information directly by loading each sector into memory and examining it with a hex-ASCII dump which is formatted very differently from the examples in the DOS Manual. Finding your way around in the hex-ASCII dump by consulting the DOS Manual can be quite a chore as I found out several months ago while trying to restore a master diskette that was somehow damaged by an accidental reset during a copy operation (even though the jacket was not notched!). The damage caused the CATALOG command to list the first file name correctly and then list garbage for the second file name and quit. I first tried RECONSTRUCT VTOC, but the disk drive made strange groaning noises I had never before heard, so I quickly abandoned that method with a reset. The next step was to examine the VTOC from a hex-ASCII dump and it seemed to be OK. However, an examination of the diskette directory revealed a collection of garbage following the entry for the first file. I had to rebuild the first sector byte-by-byte using the DOS Manual as a guide and kept thinking, "There must be an easier way". The accompanying table is an outgrowth of that desire to find an easier way.

There are several good utilities around, including DAN'S DISK UTILITIES, DISK FIXER and DISK ZAP, which will allow you to read in a specified sector of the directory, examine it with a hex-ASCII dump, selectively change the sector data and then write the modified sector back out of the disk. The format of all the hex-ASCII dumps I have seen involves 32 lines, each line beginning with the address (in hex) of the first byte in the line, followed by the hex codes of eight bytes of the sector, followed by the ASCII representation of the eight bytes. When you examine a directory sector using such a program, the only part of the sector which will make much sense at first is the ASCII representation of the file names, but the other nonprinting bytes contain very important information. The table at the end of this article was designed to facilitate interpreting a typical directory sector from a hex-ASCII dump, and it has since proven so useful in my work that I thought I would make it available to the Call —A.P.P.L.E. membership. I keep my copy on the first page of my collection of "DOS Notes".

The directory of a normal 3.3 diskette begins with track $11 (17) sector $0F (15) and continues *backwards* to sector $0E (14), $0D(13),. . . to sector $01(1). The first 11 bytes of each sector of the directory do not pertain to any particular file. Byte $00 is not used. (All unused bytes are coded "XX" in the accompanying table.) Bytes $01 and $02 ("LT" and "LS") give the track and sector, respectively, of the block where the directory is continued from the current sector. If you dump track $11, sector $0F, these bytes should be $11, $0E. If bytes $01 and $02 are both $00, the CATALOG command will list only the files included up through the current sector, and files with entries in other sectors of the directory will be unavailable to DOS. Bytes $03 through $0A are apparently padded with zeros so that the last byte of the current sector will turn out to be the "end mark" code for the last (seventh) file in the sector.

The black lines across the table mark the beginning and ending of a file's entry in the directory . The first two directory entries for each file are symbolized in the table by "TT" and "TS". They give the track and sector location of the beginning of the track-sector list for the file. The track/sector list is just a list of the tracks and sectors where the contents of the file are stored. A large file may be spread out across numerous noncontiguous sectors if it is SAVEd onto a diskette containing many other small files and having a history of numerous file DELETEions. In such cases, the drive will make numerous sliding noises as it bounces back and forth between these widely separated tracks during a LOAD. If the same file were SAVEd onto a diskette that has not had a history of DELETEions, the track/sector list will consist mostly of numerically contiguous sectors and access time may be considerably improved. An entire disk may be cleaned up this way copying it with program FID from your DOS 3.3 master disk, rather than using the other copy programs available, which usually copy the disk as is, sector by sector rather than file by file. If "DeMuffined", FID is usable in a 3.2 DOS environment.

"FT" is the file type (and lock code). If the first hex digit is $0, the file is unlocked; if it is $8, the file is locked; if it is $1, it is an "R" or relocatable binary file such as can be produced by the DOS Tool Kit. The second digit is file type with 0=text, 1=Integer, 2=Applesoft, 4=binary, 8=the mysterious "S" file (Does anyone know anything about "S" files??), and 16 ($10) the "relocatable" binary file used by the DOS Tool Kit. If you are tempted to try converting a file from Integer to Applesoft by changing "FT" from $?1 to $?2, recall that while listings of Integer and Applesoft programs may look very similar, the programs are very different in terms of their organization in memory. If you change the "FT" of an Integer program from $?1 to $?2,

EXAMINING DIRECTORY SECTORS

the CATALOG command will list the file as Applesoft, and LOADing the program will automatically switch on the Applesoft interpreter if you have both BASICs in ROM, but the program will arranged improperly in memory and will be *totally* useless. Try it! It is, however, possible to change any file's type code to $00 and it can then be read as a text file, but reading will stop at the first occurrence of a null byte ($00) anywhere in the file. This eliminates a FP file.

"fn" stands for the hex code of one letter of the file name. Names shorter than 30 characters are appended with trailing blanks (hex code $A0). You can enter inverse, flashing, or control characters into a file name using the hex codes from page 15 of the new Apple][Manual.

"SC" stands for the number of sectors the file takes up, but this number is apparently not used by DOS for anything but the CATALOG command. The track/sector list will still be read by DOS all the way to the end when you LOAD a file, regardless of the value of "SC".

"EM" is normally $00 and serves as an end mark for the file's entry in the directory. The DOS 3.2 Reference Manual explains on page 130 that the DELETE command does not actually delete the file from the diskette, but merely marks the file's entry by changing the "TT" value to $FF. The file name and the data are still there unless you subsequently SAVE another file, in which case the old file name is replaced by the new file name, and at least part of the old file data may be overwritten. At the bottom of page 130, the DOS 3.2 Manual explains that when a file is DELETEd, the value of "TT" is transferred to "EM" for safe keeping. *This is wrong!* Check for yourself. DELETE a file and then note that its "EM" value is still $00. What actually happens is that the value of "TT" is transferred to the position of the last character of the file name (the last "fn"). Then if you run a file-restoring program like IAC's DISK HELPER, the value stored in the last "fn" is transferred back to "TT" and the file is suddenly restored! In the process, the last character of the file name gets converted to a blank (hex code $A0) regardless of what it was previously. I have circled the last "fn" position of each entry in the table to serve as a reminder of its unique role. This error in the DOS 3.2 Manual was corrected in the DOS 3.3 Manual with no comments about the change.

I hope you find the directory guide useful. Make a copy of one of your diskettes with many different kinds of files on it and begin experimenting around using one of the sector access programs available. You would be amazed at what you can learn about DOS by just messing around with the directory sectors!

SYMBOLIC GUIDE TO HEX DUMP OF ONE SECTOR OF DISKETTE DIRECTORY

	$0	$1	$2	$3	$4	$5	$6	$7
$00	XX	LT	LS	XX	XX	XX	XX	XX
$08	XX	XX	XX	TT	TS	FT	fn	fn
$10	fn	fn	fn	fn	fn	fn	fn	fn
$18	fn	fn	fn	fn	fn	fn	fn	fn
$20	fn	fn	fn	fn	fn	fn	fn	fn
$28	fn	fn	fn	(fn)	SC	EM	TT	TS
$30	FT	fn	fn	fn	fn	fn	fn	fn
$38	fn	fn	fn	fn	fn	fn	fn	fn
$40	fn	fn	fn	fn	fn	fn	fn	fn
$48	fn	fn	fn	fn	fn	fn	(fn)	SC
$50	EM	TT	TS	FT	fn	fn	fn	fn
$58	fn	fn	fn	fn	fn	fn	fn	fn
$60	fn	fn	fn	fn	fn	fn	fn	fn
$68	fn	fn	fn	fn	fn	fn	fn	fn
$70	fn	(fn)	SC	EM	TT	TS	TF	fn
$78	fn	fn	fn	fn	fn	fn	fn	fn
	$8	$9	$A	$B	$C	$D	$E	$F

	$0	$1	$2	$3	$4	$5	$6	$7
$80	fn	fn	fn	fn	fn	fn	fn	fn
$88	fn	fn	fn	fn	fn	fn	fn	fn
$90	fn	fn	fn	fn	(fn)	SC	EM	TT
$98	TS	FT	fn	fn	fn	fn	fn	fn
$A0	fn	fn	fn	fn	fn	fn	fn	fn
$A8	fn	fn	fn	fn	fn	fn	fn	fn
$B0	fn	fn	fn	fn	fn	fn	fn	(fn)
$B8	SC	EM	TT	TS	FT	fn	fn	fn
$C0	fn	fn	fn	fn	fn	fn	fn	fn
$C8	fn	fn	fn	fn	fn	fn	fn	fn
$D0	fn	fn	fn	fn	fn	fn	fn	fn
$D8	fn	fn	(fn)	SC	EM	TT	TS	FT
$E0	fn	fn	fn	fn	fn	fn	fn	fn
$E8	fn	fn	fn	fn	fn	fn	fn	fn
$F0	fn	fn	fn	fn	fn	fn	fn	fn
$F8	fn	fn	fn	fn	fn	(fn)	SC	EM
	$8	$9	$A	$B	$C	$D	$E	$F

"EM" = End Mark: Last byte of directory entry for that file. Always equal to $00.

"FT" = File Type: First digit is $0 if unlocked, $8 if locked, $1 if a Relocatable file. Second digit is file type with $0=text, $1=Integer, $2=Applesoft, $4=binary, $8=special, and $10=relocatable.

"fn" = File Name: These bytes are the hex codes of the characters in the file name. If the file is DELETEd, the last fn byte (circled) is used to store the old value of "TT".

"LT" = Link Track: Track where directory is continued. This is usually $11 (or $00 indicating no more links).

"LS" = Link Sector: Sector of track "LT" where directory is continued.

"SC" = Sector Count: The number of sectors occupied by the file.

"TT" = Track/sector list Track: Track where track/sector list of this file starts. Converted to $FF by DELETE.

"TS" = Track/sector list Sector: Sector of "TT" where track/sector list of the file starts.

" " = Boundaries between the entries for successive files in the directory.

For A.P.P.L.E. Members Only

A.P.P.L.E. SPECIAL

You Get:

BIG MAC.LC	**$28.50**
Assembly Lines	**$16.50**
Using 6502	**$17.00**
	$62.00

MEMBER PRICE: $52.50

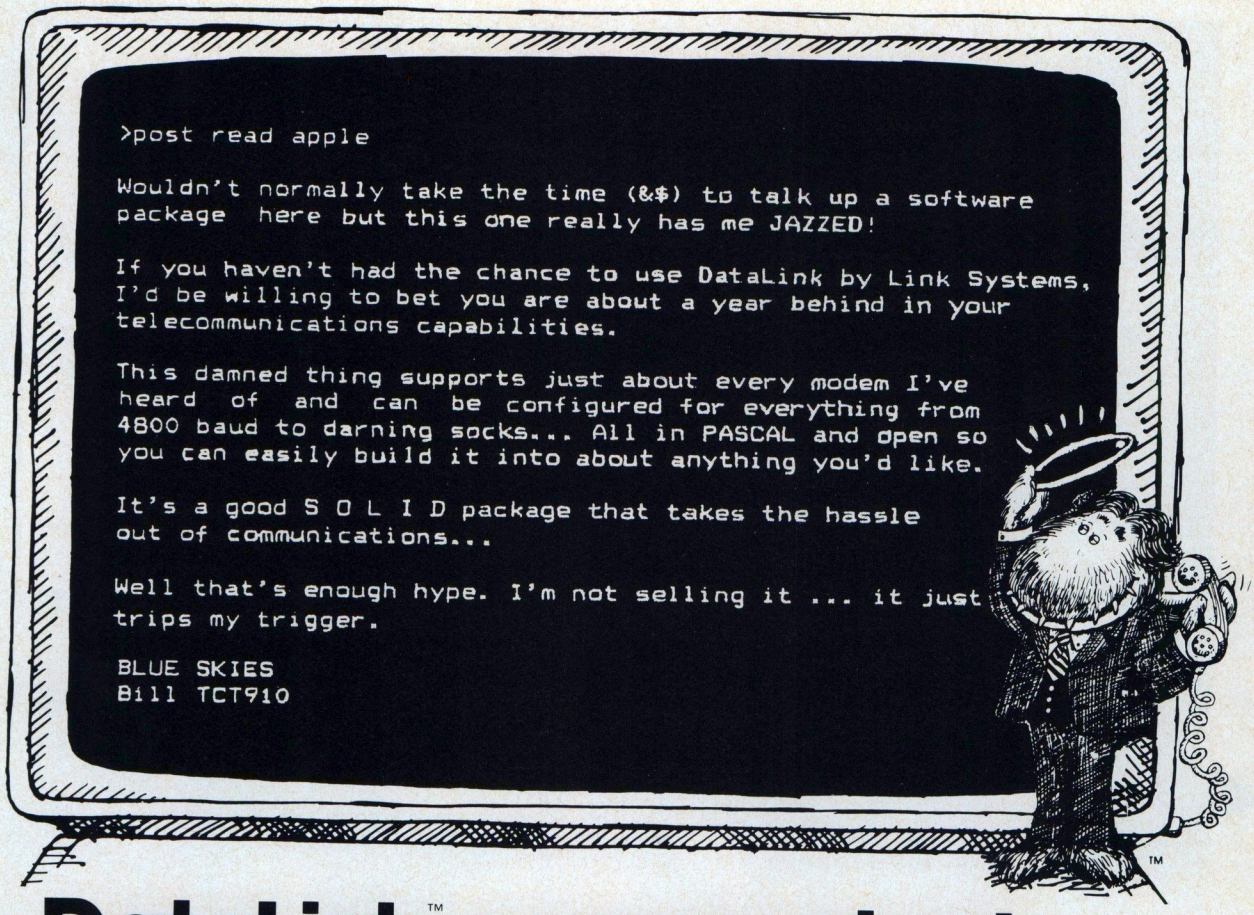
Must be ordered by 8/31/82

A.P.P.L.E. Orders
304 Main Ave. S., Suite 300
Renton, WA 98055
(206) 271-4514

Overseas shipping weight: 29 oz.

Washington state residents add 6.5% sales tax.

All foreign orders (including Canada) will require additional postage and shipping.



DataLink™ communicates.

We recently came across the above message on the Source network. Seems that quite a few people are discovering what we already know—nothing communicates like DataLink.

Sophisticated yet simple to use, DataLink from Link Systems can be applied to a broad variety of functions. You can access timesharing systems, data bases and Apple bulletin board systems. Transmit electronic mail. Send and receive Pascal files. Access university computer systems. Even "chat" with friends.

Just one keystroke sets environment, dials you in and logs you on—or automatically redials if you need to try again. By using sophisticated main-frame error-checking protocols, DataLink makes sure your transmissions are virtually error-free.

DataLink saves time too. You can send any size file you want at 4800 baud computer to computer. Or up to 1200 baud over phone lines.

DataLink works with most any card or modem available for your Apple. Send us the coupon and we'll send you all the details. But to really appreciate the power and ease of this new system, see your computer dealer.

Word's getting around—nothing communicates like DataLink.

I want DataLink™ to trip my trigger!

☐ **Please rush me more information.**
Apple II ___ Apple III ___ IBM ___

☐ **Dealer inquiry**

☐ **Send me (quantity)** _____
DataLink™ for Apple II at $99.95 each. CA residents add 6% tax. Shipping in U.S.A. $3.00 for first copy, 50¢ each additional copy. Foreign, add an additional $10 shipping and handling charge per order.

Payment by
☐ VISA ☐ MasterCard ☐ Check

Credit Card # _____
Expiration date _____
Interbank #(MC only) _____
Signature _____
Name _____
Company _____
Address _____
City _____
State _____ Zip _____

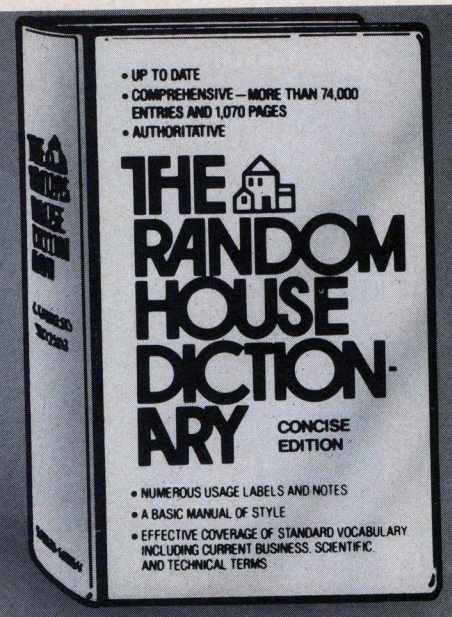
Mail to: Link Systems
1640 19th St
Santa Monica, CA 90404
(213) 453-1851
CA

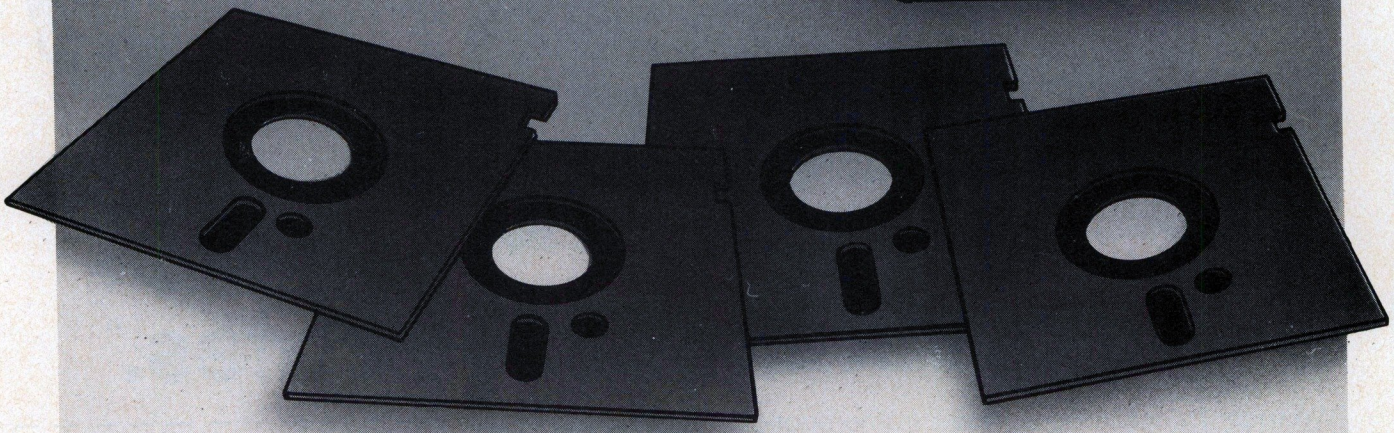
Apple is a registered trademark of Apple Computer Co., Inc.

LINK SYSTEMS™

Simply powerful software that links it all together™

WE'VE CHANGED MORE ON THE APPLE SPELLER THAN JUST THE NAME





You get 85,000 words on disk with THE SENSIBLE SPELLER and a Random House dictionary to boot!

THE SENSIBLE SPELLER™ is the first COMPLETE spelling verification program for the Apple Computer. NO other spelling program includes as many options, or has a dictionary (on diskette) as large as ours, and includes a well known dictionary, too. THE SENSIBLE SPELLER now features the complete CONCISE EDITION of the RANDOM HOUSE® DICTIONARY. If you question the definition, pronunciation, or spelling of any word, you can simply look it up in the supplied hardcover dictionary.

Four versions of THE SENSIBLE SPELLER are available. There is no need to MUFFIN or transfer your files from one operating system to another.

- The *DOS 3.3 version* works with Applewriter (I or II), Apple Pie, The Correspondent, Executive Secretary, Letter Perfect, Magic Window, Screenwriter (formerly Superscribe), Text Editor, TXT/ED, Write-On, Word Power, or any other word processor, editor, or program that generates standard DOS 3.3 TEXT or BINARY files.
- The **Super Text™ version** works with Super Text, Super Text II, and the new 80-column version of Super Text.
- The **CP/M™ version** works with WordStar, ED, Magic Wand, Type Master, and any other word processor that does not compact text.
- The **Pascal™ version** works with the Pascal Editor, and Prose.

Numerous options are provided throughout the program to enable you to completely control all activities of THE SENSIBLE SPELLER. These include the ability to ignore both formatting commands and control codes, multiple options for the action taken with each misspelled word, and the use of multiple dictionaries. An alphabetical listing of misspelled, unknown, and/or all words in your document along with usage frequencies may be listed. A verification mode is provided to allow you to examine and dispense with misspelled words while viewing them in the actual context in which they appeared in your file. When you encounter an unknown word, you can even search the dictionary for the correct spelling (using wildcards). In the near future, Sensible Software will be releasing well known legal and medical dictionaries to complete this professional package.

THE SENSIBLE SPELLER package includes:

- an easy to read instruction manual
- two copies of THE SENSIBLE SPELLER program
- a main dictionary diskette, containing 45,000 Of the most commonly used English language words (there is room to add approximately 10,000 or your own words)
- a supplementary dictionary diskette, containing the remaining 40,000 words in the CONCISE EDITION of the RANDOM HOUSE DICTIONARY
- the hard cover RANDOM HOUSE DICTIONARY, CONCISE EDITION

THE SENSIBLE SPELLER requires an Apple™ II/Apple™ II+ equipped with 48K, DOS 3.3, and 1 or 2 disk drives. Two disk drives are required to delete or add words to the dictionary. The price is only $125.00.

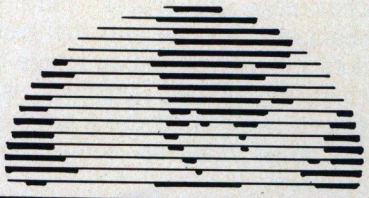
Sensible Software™

6619 Perham Drive, Dept. C
West Bloomfield, Michigan 48033
(313) 399-8877
Visa and Mastercard Welcome.
Add $1.25 postage and handling per program.
For a complete catalog, send $1.00, refundable with your first purchase.

• APPLE is a registered trademark of APPLE Computer Company • CP/M is a registered trademark of Digital Research Corp. •
• SUPER TEXT is a registered trademark of Muse Software • APPLE PASCAL is a registered trademark of U.C.S.D. •
RANDOM HOUSE and the House design are registered trademarks of Random House, Inc. Copyright 1982 Random House, Inc.

Ins and Outs

Tom Little



Part Two

IN the last article we talked about the way in which the Apple uses its internal address bus to control input and output operations. Methods of controlling the address bus from BASIC and Assembly Language were shown, and examples were given illustrating the Apple's "on-board I/O" (keyboard, speaker, game paddles and so on). The I/O addresses not covered last time all cause signals to be sent to the peripheral card slots.

To understand about peripheral cards, we should digress first into software, and leave the hardware waiting. The Apple comes with standard programs in permanent memory (ROM) for doing input from the keyboard and output to the TV screen or video monitor. These built-in programs are part of a larger program called the Apple System Monitor, or just Monitor for short.

Let's first consider the problems of putting a character on the video display. The video circuitry gets its information from the Apple's temporary (RAM) memory. When the Apple is displaying the primary page of text (which is nearly all the time), all the information on the screen is kept in memory locations $400 to $800. To output a character to the screen, all you need to do is put the numeric code (see page 15 of the Apple Reference Manual) for that character in the proper memory location in the $400-$800 range. This would be a very simple thing to do if the positions on the screen corresponded to the locations in memory in a nice, sequential manner. You might expect, for example, that the upper left corner of the screen would be an image of memory location $400, the next character position would be $401, and so on. Since there are 40 columns of text on the Apple screen, you would expect that the first position of the second line would be $400 + $28 (= 40 decimal), or $428. Alas, it is not so. All the positions in a single line of text correspond to memory locations in sequence, but the sequence is broken when you move from one line to the next.

The primary page of text is stored in locations $400 - $800.

Screen memory is not entirely sequential.

The TEXT MEMORY DEMO program (listing number 1) illustrates this. When it runs, it shows you the memory address which corresponds to each position on the text screen. Some of the memory locations in the range $400 to $800 do not show up on the screen at all! Notice that $400 (hex) is the same as 1024 (decimal).

The GRAPHICS MEMORY DEMO program (listing number 2) shows the same information, but for the Hi-Res graphics screen instead. Notice that each memory location for the graphics screen shows up as just a narrow line on the video display.

Listing 1

```
]LIST
10  REM
TEXT MEMORY DEMO BY TOM LITTLE

 CALL -A.P.P.L.E. : JULY 1982

50  TEXT : HOME : VTAB 24
100 IS = 32: REM   INVERSE SPACE
110 NS = 160: REM  NORMAL SPACE
120 T1 = 1024: REM  BEGINNING OF
     TEXT MEMORY
130 T2 = 2048: REM   END OF TEXT
     MEMORY
140  REM
150  FOR LOC = T1 TO T2
160  POKE LOC,IS
170  HTAB 1: PRINT "LOCATION #";L
     OC;
180  FOR PAUSE = 1 TO 100: NEXT P
     AUSE
190  POKE LOC,NS
200  NEXT LOC
210  END
```

Listing 2

```
]LIST
10  REM
HI-RES MEMORY DEMO BY TOM LITTLE

  CALL -A.P.P.L.E. : JULY 1982

50  HGR : HOME : VTAB 24
100 WHITE = 255: REM  ALL PIXELS
     LIT
110 BLACK = 0: REM  NO PIXELS LIT

120 G1 = 8192: REM  BEGINNING OF
     HI-RES GRAPHICS MEMORY
130 G2 = 16384: REM  END OF HI-RE
     S GRAPHICS MEMORY
140  REM
150  FOR LOC = G1 TO G2
160  POKE LOC,WHITE
170  HTAB 1: PRINT "LOCATION #";L
     OC;
180  FOR PAUSE = 1 TO 10: NEXT PA
     USE
190  POKE LOC,BLACK
200  NEXT LOC
210  END
```

This is because when text is displayed, all we need to store in memory is a code for the character. A character generator chip then changes this code into the group of dots which make up an image of the character. When graphics is displayed, we cannot use a code; every dot must take up some room in memory.

So it is not an easy task to print a string of characters in successive screen locations, which is what we must do if we want to display a message on the screen.

Don't lose any sleep over it though, because there is a Monitor routine called COUT1 which does just that. An Assembly Language programmer can load the accumulator register with the code of the character he wants to print out on the screen and then call COUT1 (JSR $FDF0). COUT1 keeps track of the

COUT1 keeps track of the next available screen location.

next available screen location for a character to be printed, and stores the code in the proper memory location for that screen location. It also understands that a carriage return character means that the output cursor (the next available screen location) should be moved to the first column of the next line, and that a bell character should beep the Apple's speaker. It also "scrolls" the screen upward if there is no more room. In short, COUT1 does everything a good screen output program should. Almost every time the Apple prints text on the screen, it uses COUT1.

The folks who designed the Apple were smart. They realized that if all output was done through COUT1, it would be very difficult to get output to go to a printer, a modem, or whatever. Also, they reasoned, if the user did not like the way COUT1 handles output, he should be able to write his own character output routine. So they invented CSW.

CSW is a location in the Apple's temporary memory ($36). This memory location contains the address of the routine which should be used for output. Ordinarily, it contains $FDF0 (the address of COUT1). Notice, by the way, that addresses are two bytes long, so CSW really includes the two locations $36 and $37. The Apple stores numbers with the last (least significant) byte in the lower memory location, so if CSW contained the address of COUT1, $36 would contain $F0, and $37 would contain $FD.

The Apple makes use of CSW by means of the subroutine COUT, located at $FDED in the Apple System Monitor. When an Assembly Language program does a JSR COUT, it jumps to $FDED, where there is an instruction that does an indirect jump through CSW. This means that the program will then continue executing at whatever address is stored in CSW, usually $FDF0. Admittedly, this is a roundabout way to get to COUT1 ($FDF0), but it allows us to intercept the out-

You could write a short program to do customized output . . .

put process. You could write a short program to do customized output, and put its address in CSW. After you did this, all calls to COUT would end up running your program instead of COUT1. Since nearly all Apple software uses COUT for output, most output would then be done through your own program.

Keyboard input works the same way. There is a monitor subroutine, KEYIN, at $FD1B, that reads the keyboard. It is a rather simple routine. It doesn't display a flashing cursor or do anything fancy. Most input is done through a different routine, RDKEY, at $FD0C. This routine displays a flashing cursor, then jumps through KSW. KSW is for input what CSW is for output. It is a location in memory ($38 and $39) which contains the address of an input routine, normally KEYIN. Because most Apple programs get their input by calling RDKEY, changing KSW changes the Apple's method of getting input.

If you're still paying attention and haven't been off daydreaming, you will realize that all of this talk about redirecting input and output sounds a lot like the IN# and PR# commands in BASIC. Well, it's true. IN# changes KSW and PR# changes CSW, so that I/O is done through the peripheral card of your choice.

Ah! Peripheral cards! It is now time to talk about the rest of Apple's special I/O addresses, which refer to peripheral cards. After all, KSW and CSW are supposed to contain the address of a program. How can a peripheral card be a program? What do PR# and IN# really do?

Nearly all of the Apple's 4096 separate I/O addresses refer to peripheral card slots. Surprisingly, though, each slot only gets three unique signals from the Apple. The names of these three signals are DEVICE SELECT, I/O SELECT, and I/O STROBE. The I/O STROBE signal goes to all the peripheral card slots, but each slot has its own special DEVICE SELECT and I/O SELECT. To be perfectly accurate, it should be noted that slot 0 doesn't get an I/O SELECT signal.

The Apple's built-in I/O addresses each do specific tasks, like reading the keyboard. What, then, do the peripheral card I/O signals do? The answer is that it depends on the peripheral card. A card can be built to use its three signals however it likes.

You have no idea what referencing a peripheral address will do . . .

You have no idea what referencing a peripheral I/O address will do unless you have documentation for that particular peripheral card which tells you the answer.

This is a good time to look at Table 1 to see which addresses produce which signals.

The DEVICE SELECT signal for each card is produced by referencing one of the sixteen addresses indicated in the table. Most peripheral cards use these addresses to get commands from the Apple. If you have a Language Card (in slot 0 I hope!), you will recognize the addresses $C080 to $C08F as corresponding to the commands you can give to the Language Card about whether it should be enabled for reading and writing and so on. All the peripheral card slots have access to the Apple's entire address bus, so when a card gets the DEVICE SELECT, it can check the address bus to find out which particular address was used to produce the DEVICE SELECT signal, and act ac-

Table 1

Apple II I/0 Addresses

ADDRESESS (HEX)	(DECIMAL)	NAME	FUNCTION
C000 to C00F	49152 to 49167	Keyboard Data	allows the keyboard to place its data on bits 0–6 of the data bus; bit 7 is on if a key has been pressed
C010 to C01F	49168 to 49183	Keyboard Strobe	tells keyboard to wait for the next keypress; should be done after each time a character is obtained
C020 to C02F	49184 to 49199	Cassette Output	toggles the cassette output (produces a "click" on a cassette recording)
C030 to C03F	49200 to 49215	Speaker Output	toggles the speaker output (produces a "click" on the Apple's speaker)
C040 to C04F	49216 to 49231	Utility Strobe	sends a pulse to the Utility Strobe connection on the game I/O port
C050	49232	Graphics mode	tells the Apple's video circuitry to display graphics on the screen, as opposed to text
C051	49233	Text mode	tells the Apple's video circuitry to display text on the screen
C052	49234	Nomix	tells the Apple's video circuitry to display graphics only (no split screen); graphics mode must be on
C053	49235	Mix	tells the Apple's video circuitry to display graphics and text together (split screen); graphics mode must be on
C054	49236	Page 1	tells the Apple's video circuitry to display the primary page (either graphics or text)
C055	49237	Page 2	tells the Apple's video circuitry to display the secondary page (either graphics or text)
C056	49238	Low Resolution	tells the Apple's video circuitry to display lo-res graphics (stored in the text pages); graphics mode must be on
C057	49239	High Resolution	tells the Apple's video circuitry to display hi-res graphics; graphics mode must be on
C058	49240	an0 off	turns off the "annunciator 0" output on the game connector
C059	49241	an0 on	turns on the "annunciator 0" output on the game connector
C05A and C05B	49242 and 49243	Annunciator 1	likewise
C05C and C05D	49244 and 49245	Annunciator 2	likewise
C05E and C05F	49346 and49247	Annuciator 3	likewise
C060	49248	Cassette Input	puts the current state of the cassette input on bit 7 of the data bus
C061	49249	Pushbuttom 0	puts the current state of the pushbutton on paddle 0 onto bit 7 of the data bus. (Don't ask why the Apple Reference Manual calls it pb1)

C062	49250	Pushbutton 1	likewise
C063	49251	Pushbutton 2	likewise
C064	49252	GC 0	puts the current state of the game controller 0 input on bit 7 of the data bus. Refer to the text on how to use this to read the game paddle dial.
C065	49253	GC 1	ditto
C066	49254	GC 2	likewise
C067	49255	GC 3	likewise
C068 to C06F	49256 to 49263		repeat of C060 to C067
C070 to C07F	49264 to 49279	GC Strobe	Resets the timing circuits for use with the Game Controller inputs.
C080 to C08F	49280 to 49295	DEVICE SELECT 0	sends the DEVICE SELECT signal to slot 0. The peripheral card in that slot can interpret these addresses however it chooses.
C090 to C09F	49296 to 49311	DEVICE SELECT 1	likewise
C0A0 to C0AF	49312 to 49327	DEVICE SELECT 2	likewise
C0B0 to C0BF	49328 to 49343	DEVICE SELECT 3	likewise
C0C0 to C0CF	49344 to 49359	DEVICE SELECT 4	likewise
C0D0 to C0DF	49360 to 49375	DEVICE SELECT 5	likewise
C0E0 to C0EF	49376 to 49391	DEVICE SELECT 6	likewise
C0F0 to C0FF	49392 to 49407	DEVICE SELECT 7	likewise
C100 to C1FF	49408 to 49663	I/O SELECT 1	sends the I/O select signal to slot 1. Ordinarily these addresses will refer to a permanent memory chip on the peripheral card, which is activated by the I/O select signal.
C200 to C2FF	49664 to 49919	I/O SELECT 2	likewise
C300 to C3FF	49929 to 50175	I/O SELECT 3	likewise
C400 to C4FF	50176 to 50431	I/O SELECT 4	likewise
C500 to C5FF	50432 to 50687	I/O SELECT 5	likewise
C600 to C6FF	50688 to 50943	I/O SELECT 6	you guessed it
C700 to C7FF	50944 to 51199	I/O SELECT 7	uh huh
C800 to CFFF	51200 to 53247	I/O STROBE	sends the I/O strobe signal to all slots. Several cards can have a permanent memory chip accessed by these addresses, but they must cooperate, since each slot does not get its own unique signal.

cordingly. So any peripheral card could do up to 16 different actions when it gets a DEVICE SELECT signal, depending on which particular number is on the address bus at the time.

This means that peripheral cards generally have to have decoding circuits ("gnomes") of their own in order to receive any reasonable number of commands from the Apple. For example, a printer interface in slot 1 might use its DEVICE SELECT addresses ($C090 to $C09F) to do things like put the printer on line, control echoing of characters on the video screen, blow up your computer, or whatever. It's all up to the interface card. Depending on how the peripheral card is designed, it might totally ignore the DEVICE SELECT signal, although this is not likely.

The next signal that goes to the peripheral cards is the I/O SELECT signal. By the way, if you're trying to figure out the significance of these names, don't bother. They had to call them something, but it's hard to name something when you don't know what it'll be used for! Each slot's I/O SELECT signal is produced by referencing any one of 256 different addresses. That's a lot of addresses. Most peripheral cards do not use the I/O SELECT signal to get commands from the Apple (which is a common use of DEVICE SELECT). Instead, many peripheral cards have a small permanent memory chip (a PROM) which is activated by the I/O SELECT signal. In this way, each of the special I/O SELECT addresses corresponds to an actual memory location on the peripheral card.

The idea behind this is that each peripheral card can have a short program which is permanently on the card, so you may not need to load in special software to use a peripheral card.

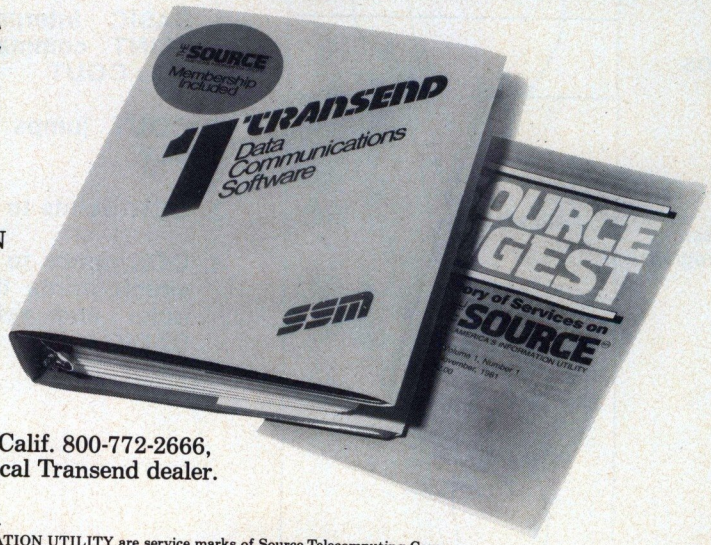
Now you can see how PR# and IN# work. When you type PR#1, BASIC puts the address $C100 in CSW ($C100 is the first address in the range which produces the I/O SELECT 1 signal). Now the next time the Apple tries to output a character by calling COUT, the program will jump to $C100. This jump puts $C100 on the address bus and activates I/O SELECT 1. We now pray that there is a peripheral card in slot 1 which can do something reasonable when it gets this signal. A printer card would almost surely have a permanent memory chip on it, so that jumping to $C100 would run the program on the memory chip, and hopefully turn the printer on and output the character.

. . . jumping to $C100 would run the program on the chip . . .

So although the BASIC manual will tell you that PR#n will send all subsequent output to a peripheral card in slot n, it would be more accurate to say that PR#n will send an I/O SELECT signal to slot n the next time output is requested. There is a difference. The disk controller card, for example, does not print a character on the disk when you say PR#6. It does, however, run the program on its own permanent memory chip (activated by I/O SELECT 6). This program boots the disk.

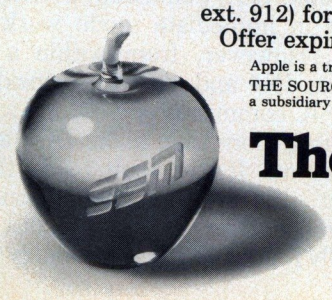
The point of all this is that although many peripheral cards in fact do perform I/O when you type PR#n or IN#n, they don't have to. Again, it is up to the designer of the card.

Apple™ users. THE SOURCE℠ and TRANSEND™. Together for the first time for only $89.



Buy our $89 Transend state-of-the-art data communications software and membership in THE SOURCE, AMERICA'S INFORMATION UTILITY℠, is included. This combination allows you to easily access one of the world's largest information services for up-to-the-minute news and sports, stock prices, travel services and much more.

An optional 260-page Source User's Manual is available for $19.95. Your dealer has complete details. Call 800-227-2400, ext. 912 (in Calif. 800-772-2666, ext. 912) for the name of your local Transend dealer. Offer expires Oct. 1, 1982.



Apple is a trademark of Apple Computer Inc.
THE SOURCE and AMERICA'S INFORMATION UTILITY are service marks of Source Telecomputing Corp., a subsidiary of The Reader's Digest Association Inc.

The Transformation People.

SSM Microcomputer Products Inc.
2190 Paragon Drive, San Jose, CA 95131

INS AND OUTS

Because each slot has its own I/O SELECT signal, several cards can be active at once. Of course there can only be one address in CSW or KSW at a time, but an Assembly Language program (or even BASIC) could call programs on different cards. More on this later.

The final signal, I/O STROBE, is sent to ALL slots whenever a number between $C800 and $CFFF is on the Apple's address bus. That means that there are 2048 addresses (that's half of the Apples total I/O addresses) that activate the I/O STROBE. The fact that this signal is sent to all the peripheral cards means that they have to come to some kind of agreement about what it means, or else disaster will ensue.

The intended use of I/O STROBE is to enable a large permanent memory chip on one peripheral card. This way, a peripheral card could have 2K memory for a program or data, which would always correspond to $C800–$CFFF, regardless of which slot the card happened to be in. But if several cards used this memory area, the Apple would be unable to determine which one to look at. To resolve the problem, the Apple designers made it clear that all peripheral cards should interpret the address $CFFF as a command to turn off their $C800 memory if they have one. One peripheral card can gain control of this address range by referencing address $CFFF to deactivate the $C800 memory on the other cards.

The mechanics of how this all works is of interest mainly to people who design peripheral cards. But the programmer must be aware that the addresses $C800 to $CFFF may be "shared" among several peripheral cards. The addresses will refer to whichever card thinks it has control. For example, the Apple SILENTYPE printer uses some of these addresses to store its parameters (margins, intensity, etc.). When you give the printer control (by "PR#1", for example), then it will make sure that addresses $C800 to $CFFF refer to memory locations on its own card. This is why you may set the SILENTYPE's parameters only after you initialize it with the PR# command.

If you're beginning to think the process of outputing a single character is astonishingly complicated, you're right. But, as they say, You ain't seen nuthin' yet. There is a powerful software life-form known as DOS (Disk Operating System, for the uninitiated), which has its claws into everything you try to do. DOS is not content to let you play around with peripheral I/O all by yourself.

DOS gets its commands either through input (when you type them from the keyboard) or through output (when your program says PRINT D$"CATALOG" or something similiar). For this reason, DOS feels that it must have control of CSW and KSW so that it can grab any DOS command as it goes through. To achieve this goal, DOS puts its own addresses in CSW and KSW, and it tries very hard to make sure you don't change them. DOS would rather look at everything first, then decide whether it should go on to the screen, printer, or wherever you had in mind. So DOS keeps its own CSW and KSW. Doing output when DOS is around consists of jumping through CSW (as usual). But CSW now points to a routine in DOS. The DOS routine looks at the character, and if it is not a DOS command, it effectively jumps through its own internal version of CSW which contains the address of the real output routine. In essence, DOS is censoring your mail.

You can see all the trials and tribulations associated with printing out a character in Figure 1.

However, PR# and IN# still work pretty much as advertized. This is because they are DOS commands, as well as BASIC commands. The DOS PR# command has the effect of changing DOS's *private* version of CSW, leaving the *real* CSW pointing into DOS. Anything you type from the keyboard is interpreted as a possible DOS command, so IN# and PR# in immediate mode work just fine. However, from within a program, IN# and PR# are considered BASIC commands and usually will do nothing more than make DOS angry with you. If you want to do things right, your program should issue the corresponding DOS command (e.g. PRINT D$ "PR#1"). Assembly Lan-

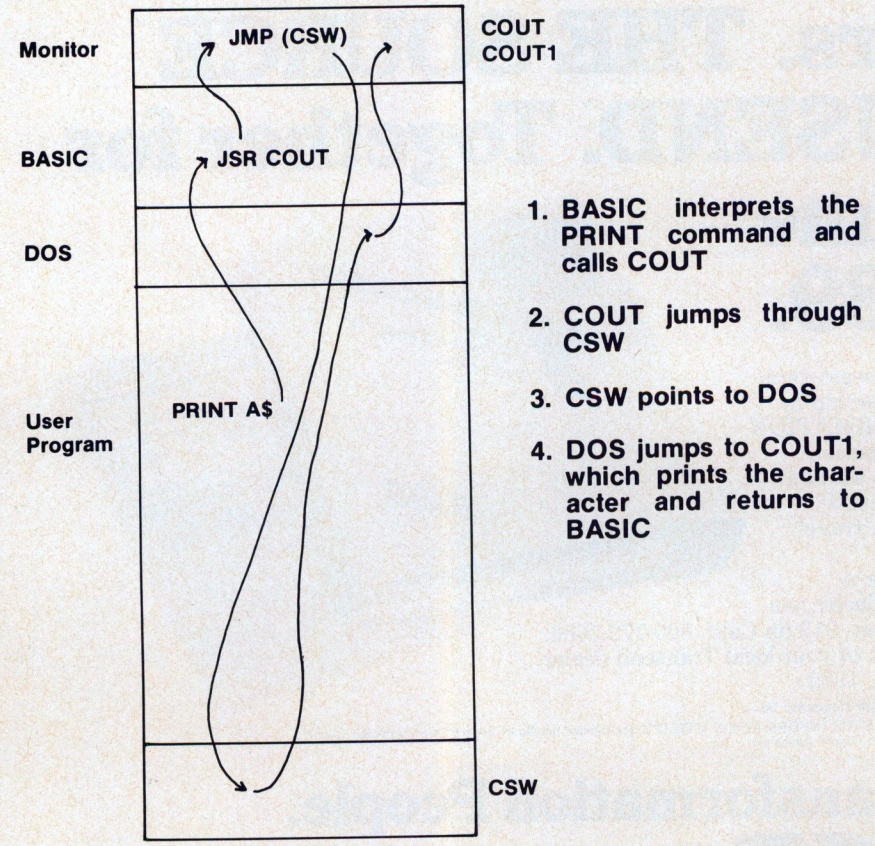


Figure 1. Character Output

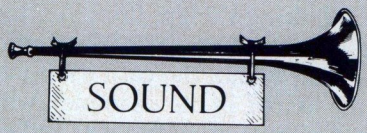
THE ROUTINE MACHINE™

SEARCH

GRAPHICS

A Versatile Programming Utility for the Apple II.

PRINT USING



Now, from the programming experts at S.D.S., an easy-to-use way of putting the POWER and SPEED of machine language routines in YOUR OWN APPLESOFT PROGRAMS!

ROUTINE MACHINE does all the work for you — no knowledge of machine language programming, whatsoever, is required. Simply choose the routine you need from an ever-growing library, and Routine Machine will effortlessly put them in your program, and all done transparently! No need to be aware of or bother with BLOAD's, HIMEM:, etc.

Best of all, with just this starter package, you'll have the routines to put High Resolution **graphics** and **sound** in your programs immediately! Also included is a versatile **print using** module to banish the "decimal point demons" forever! To round out the package, we've also included powerful **search** and **sort** routines (for single dimension arrays; Search: 1000 elements in 1 second, Sort: 1000 elements in 90 seconds), and a number of other often-needed routines as well (30 routines in all).

Additional library disks titled **"Ampersoft Program Library"** are already available.

Some of the other routines in The Routine Machine (plus others not listed) are:

SWAP: Swaps two string or numeric values.

TEXT OUTPUT: Prints with no "word break" on screen.

STRING OUTPUT: Input any string, regardless of commas, etc.

ERR: Stack fix for Applesoft ONERR handling.

GOTO, GOSUB: Allows computed statements. Example: **GOTO X * 5** or **GOSUB X * 5.**

BLOAD: Load any binary file 5 times faster than normal. Hi-Res pictures load in under 2 seconds.

RESET HANDLER: Treats RESET with ONERR; or will RUN or reboot disk.

HI-RES ASCII: Character set for mixing text Hi-Res graphics.

TURTLE GRAPHICS: Versatile Hi-Res graphics routines for easy drawing of Hi-Res figures.

OUR GUARANTEE

IF YOU DON'T SAVE MORE THAN THE PURCHASE PRICE OF 'ROUTINE MACHINE' IN YOUR OWN PROGRAMMING TIME IN THE FIRST 30 DAYS YOU OWN IT, SIMPLY RETURN IT FOR A COMPLETE REFUND, NO QUESTIONS ASKED!

southwestern data systems™

P.O. BOX 582 • SANTEE, CALIFORNIA 92071 • TELEPHONE: 714/562-3670

guage programmers can also keep DOS happy by simulating the DOS PR# command. This is done by loading the real CSW with the appropriate value (say $C100 to simulate PR#1), then doing a JSR $3EA. This subroutine copies the real CSW and KSW into DOS's personal versions, and then DOS puts its own addresses back into the real CSW and KSW.

The programmer working under DOS has basically two choices: let DOS take care of it all for you (the DOS PR# and IN# commands), or else totally disconnect DOS from CSW/KSW by putting his own addresses there (the BASIC PR# and IN# commands). In order to do the latter, it is necessary to change both CSW and KSW before any I/O takes place. Unless you cut off DOS at both places it will reconnect itself and confound your best attempts to redirect I/O. nothing will do but a stake through the heart!

We can make use of DOS to some advantage, though. We can use several peripheral cards at once if we call them directly rather than through CSW/KSW. Remember that a printer interface's output routine is located in a permanent memory chip on the card. If the card is in slot 1, then this routine is in the $C100 to $C1FF memory range. So we can call it directly: JSR $C100.

There is a small problem, though. $C100 is indeed the address that is placed in CSW when you do a PR#1. But most peripheral cards need to do some initialization work the first time they're called. After it does its initialization, it then changes CSW to be the address of just the output routine, sans initialization. This way, the next time COUT is called, the initialization part is skipped.

A user program, then, should not call $C100 for every character, but instead only the first time. You can find out where the "real" output routine is located simply by looking at CSW after the first character has been sent. It's best to do this without DOS bothering you.

Program listings 3 and 4 show a program which calls a SILENTYPE printer directly, without going through CSW ("PR#"). Listing 3 is the Applesoft program which sets things up, and listing 4 is the Assembly Language routine that actually does the output. They need to be run together. This program prints the same message on the screen, on the SILENTYPE printer, and into a DOS text file, simultaneously. It does this by telling DOS to open the text file. After this, DOS will take care of all output through COUT and CSW, sending it to the file. But the Assembly Language routine also sends the character to COUT1, which prints it on the screen, and to $C107, which prints it on the printer. Do not, I repeat DO NOT try to run this program with a printer other than a SILENTYPE in slot 1. This address, $C107, is the address of the SILENTYPE's output routine. Other interface cards cannot be expected to have an output routine in this same location. The PR#1 command in the main program is necessary to allow the printer to do its initialization.

You can call any peripheral card without going through the IN#/PR# business, but you must be careful of the following things: first, you must find out the addresses of the routines you want within the card itself (such as the $C107 in the example), which is not always an easy task; and second, you should not reference the locations $C800 through $CFFF, since the cards share these addresses and you may not know which card you're referring to at any given time.

We've now covered ALL of the Apple's special I/O addresses ($C000 to $CFFF), and taken some excursions along the way. Admittedly, there hasn't been much in these articles that couldn't be found in the Apple manuals (the Reference Manual in particular). However, I've noticed it is rather difficult to get very far in the Apple Reference Manual if you're not up on flip-flops and pinouts.

The I/O features of the Apple are some of its strong points. Because every peripheral card has all those addresses associated with it, designers can make cards that communicate with the Apple in an easy, straightforward manner. The Apple undoubtably owes much of its success to the fluency of its communication with the outside world.

Listing 3

```
]LIST
10  REM
MULTIPLE OUTPUT DEMO : TOM LITTLE
  CALL -A.P.P.L.E.   : JULY 1982

70  REM   (CALLER PROGRAM)

100 D$ =  CHR$ (4)
110   PRINT D$"PR#1": REM  INITIAL
      IZE PRINTER
120   PRINT D$"PR#0": REM  BUT TUR
      N IT OFF
130   REM
200   HOME : PRINT "MULTIPLE OUTPU
      T DEMO"
210   PRINT : PRINT "WARNING - USE
       ONLY WITH SILENTYPE": PRINT
      "PRINTER INSTALLED IN SLOT #
      1 -": PRINT "SEE ARTICLE TEX
      T FOR MORE INFORMATION"
220   PRINT
230   PRINT D$"OPEN MULT OUT TEXT"

240   PRINT D$"WRITE MULT OUT TEXT
      "
250   CALL 768
260   PRINT D$"CLOSE"
```

Listing 4

```
]ASM
                ********************************
                *                              *
                *     MULTIPLE OUTPUT DEMO     *
                *                              *
                *        BY  TOM LITTLE        *
                *                              *
                * Call -A.P.P.L.E. : July 1982 *
                *                              *
                *     (Output subroutine)      *
                *                              *
                ********************************
                *
                         ORG  $300
                *
```

```
                        15    PRINTOUT EQU   $C107       ; for SILENTYPE in slot #1 ONLY
                        16    COUT     EQU   $FDED
                        17    COUT1    EQU   $FDF0
                        18    CH       EQU   $24
                        19    CV       EQU   $25
                        20    *
0300: A0 00             21             LDY   #00
0302: A5 25             22             LDA   CV
0304: 48                23             PHA
0305: A5 24             24             LDA   CH
0307: 48                25             PHA
                        26    *
                        27    * It is necessary to save the coordinates of
                        28    * the cursor (CH and CV) so that COUT1 does
                        29    * not become confused
                        30    *
0308: B9 2B 03          31    LOOP     LDA   MESSAGE,Y
030B: 20 07 C1          32             JSR   PRINTOUT
                        33    *
                        34    * send the character to the printer
                        35    *
030E: 20 ED FD          36             JSR   COUT
                        37    *
                        38    * then to the file via CSW
                        39    *
0311: 68                40             PLA
0312: 85 24             41             STA   CH
0314: 68                42             PLA
0315: 85 25             43             STA   CV
0317: B9 2B 03          44             LDA   MESSAGE,Y
031A: 20 F0 FD          45             JSR   COUT1
                        46    *
                        47    * then to the screen
                        48    *
031D: A5 25             49             LDA   CV
031F: 48                50             PHA
0320: A5 24             51             LDA   CH
0322: 48                52             PHA
0323: C8                53             INY
0324: C0 41             54             CPY   #65
0326: D0 E0             55             BNE   LOOP
0328: 68                56             PLA
0329: 68                57             PLA
032A: 60                58             RTS
032B: D4 C8 C9
032E: D3 A0 CD
0331: C5 D3 D3
0334: C1 C7 C5
0337: A0 D7 C9
033A: CC CC A0
033D: C2 C5 A0
0340: D3 C5 CE
0343: D4 A0 D4
0346: CF                59    MESSAGE  ASC   "THIS MESSAGE WILL BE SENT TO"
0347: 8D                60             HEX   8D
0348: D4 C8 C5
034B: A0 D3 C3
034E: D2 C5 C5
0351: CE AC A0
0354: D4 C8 C5
0357: A0 D0 D2
035A: C9 CE D4
035D: C5 D2 AC
0360: A0 C1 CE
0363: C4 A0 C1
0366: A0 C6 C9
0369: CC C5             61             ASC   "THE SCREEN, THE PRINTER, AND A FILE"
036B: 8D                62             HEX   8D

--End assembly--     108 bytes     Errors: 0
```

DON'T BLOW YOUR BUCKS ON
Locked-Up Software!

Beagle Bros Apple Utilities are Listable, Backup-able Customizable and Compatible with Normal Apple DOS.

NEW!

Apple Mechanic

Shape Writer/Byte-Zap Utility

by Bert Kersey

Another hot multiple-utility disk—Nine useful, listable, copyable & customizable programs—

SHAPE EDITOR: Put professional hi-res animation in your programs. Keyboard-draw any shape & let your Apple write a shape table & store it on disk. Design large/small custom typefaces too, with special characters. Many fonts on disk. LIST-able demos show how to use shapes to animate games, displays, and CHARTS & GRAPHS. A valuable time-saving utility/learning tool.

BYTE ZAP: A MUST utility. Rewrite any byte on a disk by loading a sector onto the screen for inspection. HEX/DECIMAL/ASCII display optional. Examine bytes via cursor control; enter hex, dec or ascii to change. Create illegal filenames, restore deleted files, change greeting program name, repair/protect disks, change DOS, examine program files. Clear illustrated instructions show how disk data is stored and how to access it.

MORE: A disk PACKED with useful music, text & hi-res tricks FOR USE IN YOUR PROGRAMS. Demo-writer, hi-res utilities and excellent, educational, entertaining documentation.

ONLY **$29.50**

- ☐ Apple Mechanic disk (48K min.)
- ☐ Beagle Bros Tip Book #5 (60 pgs.)
- ☐ Peeks, Pokes & Pointers Chart

Tip Disk #1

by Bert Kersey

100 programs from Beagle Bros' Tip Books 1, 2, 3 & 4— Dozens of tricks to make your Apple do things it's never done! All 100 programs are listable, copyable and changeable; each teaches another fascinating Apple programming technique.

ONLY **$20.00**

- ☐ Tip Disk #1 on disk (32K or 48K)
- ☐ Peeks, Pokes & Pointers Chart

(Note: No tip book with this disk)

Alpha Plot

Hi-Res Graphics/Text Utility

by Bert Kersey & Jack Cassidy

Here are a few of Alpha Plot's useful features. Compare with others on the market—

HI-RES DRAWING: Create hi-res pictures & charts with text, on both pages; all APPENDABLE TO YOUR PROGRAMS. Optional Xdraw cursor (see lines before drawing). Mix colors & Reverse (background opposite). Circles, Boxes, Ellipses; filled or outlined. COMPRESS HI-RES TO 1/3 DISK SPACE. Relocate any portion of an image anywhere on either page. Superimpose too & convert hi-res to lo-res for colorful abstracts!

HI-RES TEXT: Beautiful upper/lower case with descenders (no hardware required). Color & reverse characters positionable anywhere (no tab limits). Professional-looking PROPORTIONAL SPACING; adjustable character height & letter spacing. Multi-directional typing for graphs!

ONLY **$39.50**

- ☐ Alpha Plot on Disk (48K min.)
- ☐ Beagle Bros Apple Tip Book #4
- ☐ Peeks, Pokes & Pointers Chart

DOS BOSS

DISK COMMAND EDITOR

by Bert Kersey & Jack Cassidy

A classic Apple utility you will ENJOY! Rename DOS commands (CATALOG can be "Cat", etc.). PROTECT PROGRAMS; any unauthorized save-attempt produces a "Not Copyable" message. Also LIST-PREVENTION & 1-key program-run from catalog. Custom catalogs: Change Disk Volume message to your title; Omit/alter file codes. Rewrite error messages: Syntax Error can be "Oops!!" or anything! Fascinating documentation included; Hours of good Apple reading!

Dos Boss's change features may be appended to your programs. Anyone using your disks (booted or not) formats their DOS as YOU designed it.

ONLY **$24.00**

- ☐ Dos Boss on Disk (32K/48K min.)
- ☐ Beagle Bros Apple Tip Book #2
- ☐ Peeks, Pokes & Pointers Chart

Utility City

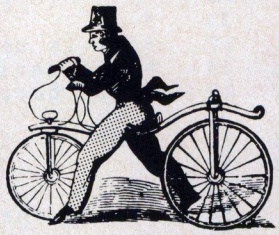
21 Useful Utilities on One Disk

by Bert Kersey

LIST FORMATTER makes properly-spaced and indented listings with page breaks; each statement on new line, if-thens & loops called out; a great de-bugger! MULTI-COLUMN CATALOG in any page-width to printer or screen. Auto-post Run-Number & last-used Date in programs. Put INVISIBLE working commands in listings. Access program lines in memory for repair & illegal alteration. Alphabetize & store info on disk. Run any program while another stays intact. Renumber to 65535. Save inverse, trick and INVISIBLE FILE NAMES. Convert dec to hex & binary, or INT to FP. Append programs. Dump text screen to printer... 21 LISTABLE PROGRAMS TOTAL!

21 PROGRAMS **$29.50**

- ☐ Utility City on disk (48K min.)
- ☐ Beagle Bros Apple Tip Book #3
- ☐ Peeks, Pokes & Pointers Chart



GOTO your Apple Dealer.

Most Apple Dealers carry our software. If your dealer doesn't, he can have our disks in his store for you within 2-3 days by calling **Beagle Bros** or **Softsel.** Or you can order directly from us. Call our Toll Free Number (below) or drop us a note with a check or your Visa/MasterCard number & expiration date.

Bonuses With Every Disk!

Poke your Apple all night long with this free reference poster! The most useable **PEEKS, POKES, POINTERS** and **CALLS,** scrounged up from every source imaginable!

Apple Tip Books too—

Each disk comes with a Gold Mine of valuable Apple information and hours of entertaining reading matter, including dozens of tips and keyboard experiments on all subjects— DOS, Copy Protection, Graphics, Shape Tables, Hardware and More. Sample programs too, such as "Programming the Reset Key" and "Copy Stoppers". Each disk comes with its own unique book.

24 Hour TOLL FREE Order Desk

MasterCard, Visa or COD Orders, call:

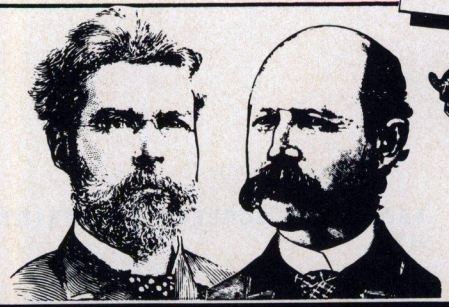
Nationwide: **800-854-2003** ext.827

California: **800-522-1500** ext.827

Alaska/Hawaii: **800-854-2622** ext.827

(Orders only please.)

COD, add $3.



OR ORDER BY MAIL:

RUSH! The disk packages checked below Plus the Tip Book & PEEKS/POKES CHART.

☐ **Alpha Plot** ☐ **Dos Boss** ☐ **Tip Disk**
☐ **Apple Mechanic** ☐ **Utility City**

(Add $1.50 Shipping, any size order. California, add 6% tax.)

NAME$= _____

ADDR$= _____

CITY$= _____

ZIP$= _____

VISA — **VISA/MASTERCARD Include Account No. & Expiration Date** — master charge THE INTERBANK CARD

Mail U.S. check, money order or Visa/MC nos. to:

BEAGLE BROS, Dept.Z
4315 Sierra Vista
San Diego, Ca 92103

(Use this coupon or separate sheet.)

All Orders Shipped Immediately

Please add $4 for shipping outside US/Canada. COD orders add $3. California residents, add 6%.

Beagle Bros™ MICRO SOFTWARE

4315 Sierra Vista, San Diego, Ca 92103
714-296-6400

"APPLE" is a registered trade mark of Apple Computer Inc.
"BEAGLE BROS" belongs to us.

A CALL —A.P.P.L.E. REVIEW:

Word Processors

Jeffrey K. Finn

EVERY time we think we've finally located all the word processors which operate on Apple computers, a couple more appear. We located several more at the West Coast Computer Faire which brings the total to almost 50 word processors to be reviewed.

"I've read the word processing reviews and I still don't know which one to buy. Please help me." This is a plea I hear many time when I answer a phone call from a fellow A.P.P.L.E. member. I truly wish there was an easy answer. Because word processors must allow for so many different situations and there are so many of them available, it is impossible to find a good fit for an individual situation without personally taking the time to learn about word processors or paying someone else to do the job for you.

The information provided in our reviews have two purposes. First, they attempt to give the reader an overview of what is available and second, they try to make the user aware of the complexity of the programs from which the user must eventually chose. The bottom line is that there is no easy answer. Time and effort will have to be spent to locate the word processor which best meets your needs and personal preferences.

Let me make it clear that I don't just hang up on the caller. Instead I try to give them a process to follow. So here in the great cookbook tradition, let me tell you what I tell my callers. Start by reviewing the *Call A.P.P.L.E. Word Processor Characteristics Checklist* to identify those features which are most valuable to you. Then take this sub-list and add anything you think of which is missing. Using your specially adapted characteristics checklist, decide which of the programs has the most of what you are looking for.

At this point, the shopper should take a test drive to get a feel for the word processor which appears to meet the need. Testing a word processor should be done using the type of documents which the user intends to produce on the system. The testing process can take from one hour to four or more hours depending on how well the program works with the particular user. The user should review at least two word processor programs in this way, in order to get a feel for how differently these programs can operate. There is no way to avoid spending this time, short of hiring someone else, if you want to have a word processor which truly meets your specific needs.

Feedback

Program developers' reactions and comments to our procedures have been extremely rewarding. One mentioned that they had found out alot about their product when they were forced to verify that our Word Processor Characteristic Checklist was correct. Still another word processor developer brought up the point that in order to implement one word processor or another some very crucial trade-offs have to be made. He mentioned the specific example of previewing on the screen what is supposed to be "exactly the way it will print" when doing a mail list merge.

What would you give to have unlimited UNITS on your Apple II*?

SOFTECH MICROSYSTEMS

If long sets of data and sometimes even short ones are to be merged into the final document, the displacement could have disasterous effects when least expected. The developer felt that a screen preview lulled the user into a false sense of security when the final printed document could be significantly altered by a single letter. Therefore, it was their decision to only approximate the final printed document in preview mode and warn the user that this was the case. A debate of these trade–offs could make for an interesting discussion.

This Month's Reviews

The *Correspondent* is a word processor whose reliable performance is the kind of program users have come to expect from Roger Wagner's Southwestern Data Systems. The *Executive Secretary* is loaded with features to meet a variety of needs including form letters and access to certain data base files.

Extra special recognition should be paid to the folks at Sofsys who bring you The *Executive Secretary*. Right on the diskette label the words "USER SUPPORT" are printed followed by a telephone number. That's as clear a statement of company policy as I've seen with respect to word processors. There is nothing like seeing something done right to make you feel good inside.

The Executive Secretary Word Processor

Traci Wynn Collins/reviewer

Writers are especially advised not to judge a book by its cover. In the case of *The Executive Secretary,* however, the notebook containing the users manual was a good indication of the great news inside. Flipping over the cover creates a professional copy holder so that a typist can easily refer to the manual while typing. Even if you aren't a secretary, everyone who has ever struggled to type corrections into a program while juggling manuals in their laps can appreciate the value of a finishing touch like that.

Glancing inside the manual is even more reassuring. The table of contents is organized as twenty five lessons and a "COMPLETE INDEX." An appendix refering to "installation" is where it won't get in the way during day to day use. Having used other microcomputer word processors, I know how frustrating it is to continually search for the real beginning of the manual behind the technical information which was only important once. When I read that the author was going to teach me how to use the program to perform 1) editing, 2) layout, and 3) production, I knew that I had finally discovered a microcomputer word processor written by and for typing professionals.

It is extremely difficult to write this review without comparisons to other microcomputer word processing systems. For it is in comparison that *The Executive Secretary* stands out, not merely useful, natural, comfortable and well thought out, but as simply amazing. I am writing this review on a commercial text processing system running on a PDP–1170 at work, where the Diablo is. I am cursing the lack of a Hayes Micromodem which is all that stands between me and preparing the text at home using *The Executive Secretary* for transmission to the 1170 using its built in electronic mail utilities. *This won't work without an Executive Secretary Program at the other end...JKF.* The program is simply better than the one at work. In the past I have used *Wordstar and W.P. Daisy* extensively on CPM systems. I am familiar with the features of *The Electric Pencil and Scriptsit* on the TRS–80. I own a copy of *Easy Writer – The Professional Package.* This program is better than any of them.

Better does not mean that *The Executive Secretary* has more "features" than any of them. Everyone of them has at least one or two features that it lacks. The CPM based systems in particular, have many more features. Better means that the features which are implimented are in a straightforward easy-to-use, easy-to-remember fashion, and better also means that this program has all of the features that a professional typist or writer is going to use day in and day out, over and over again.

Programmers who are looking for an editor, and who desire to while away their hours fiddling with their editor's code – "improving" it, or adding their own "custom" enhancements are advised to buy a different system. *The Executive Secretary* is simply not for them. While Sofsys, Inc. promises to provide some technical information upon request, so that someone like myself could solve their modem problem, there is no source code, and no advertised support for that type of activity. This program is not a general purpose solution to every problem that anyone might ever have on a computer. It is a nicely designed, completely self contained tool, for those people who have a text problem.

The Executive Secretary system includes an internal "cardfile" or database management system, an "electronic mail" system, and all of the utilities necessary for disk management such as formatting and copying. It provides a total environment for text management, gently prompting and guiding the user from the moment the Apple is turned on and the program disk autostarts. One important consideration for the prospective purchasers is that these are standard, not extra cost features. Since I lack the appropriate modem, I was unable to test the communications facilities. I will briefly discuss how the other subsystems are used, followed by a section on the disadvantages of the system, and a summation.

There are two distict functional modes in the editor: typing mode and editing mode. The creation of text is accomplished in the typing mode. Typing under the *Executive Secretary* works just like normal typing on a typewriter. Corrections in the typing mode are limited to destructive backspacing. The system does generate its own carriage returns and word wrap and it is fast. There is no need to slow your typing down so that the program can catch up with you.

One distinctive feature which can be unsettling unless you really did read your manual first, is that at the bottom of the page the typist is presented with a new blank page to type on with a single context line at the bottom. Nothing unusual about that to a typist, but if you are expecting the text to continue to scroll it is a shock.

Entering edit mode is accomplished via the ESC key. Once in edit mode, over 90 percent of all of the commands are single keystroke entries. For example, pushing the M key causes a marker to be placed in text at the current cursor position which marks the beginning or the end of a block. Likewise, pushing the Ctrl–U in typing mode marks the beginning or the end of an Underline. The cursor is restricted to the page it is on, but there are other, single key commands to move from page to page, to the end of the file, etc. There are very few two keystroke commands, and no three or more keystroke commands in the editor.

Although much of your justification is performed while you watch during text entry, and a paragraph can be cleaned up before your eyes after editing, *The Executive Secretary* draws a fairly clear distinction between editing and layout. The user can specify broad layout guidelines to cover the entire document at print time, but almost every conceivable command and flexibility is permitted to be locally exercised through commands which are typed into the body of the text. One example would be >cn which turns on centering until you type in >co which turns it off. These commands are placed on a line of their own and do not appear in the final document.

The value of this clear distinction between editing and layout really does not become obvious until you can see the power the system has to conditionally modify text at print time. You are not only able to merge data from a file such as a mailing list to get the name and address, but you can print specific areas of text based upon the value of a variable in the data file. For example, if you declare a title variable in one of your card files to contain such information as Ms., Miss, Mrs. or Mrs., you could branch on those values and send a different letter to unmarried, but socially conservative women than you would send to the men on your mailing list.

In addition to conditional printing, the system will keep track of numbered items you might refer to such as slides or illustrations finally assigning numbers to each of them at print time, once you are sure they are all there, as well as indexing and abbreviations. All of these features are implemented through embedded commands which will disrupt the visual impact of the text in your file while you are editing it. You can preview the document on the screen, but you do that by printing to the screen, not in the editor.

The data management facilities of the program are fairly powerful. In addition to the creation and recovery of text files organized as cards with lines rather than as records with fields, the system can create search and sort indexes on any of the lines for the purposes of using a subset or a specific order. It can also read the files that are produced by most of the popular data mangagement systems sold for the Apple, such as CCA, D.B. Master Utility Pack, Visifile, Information Master (most Dif files), or the Data Factory. Up to one hundred fields or lines per card or record can be inserted into images of your text file. If it is appropriate, a line in the built–in card file can be summed and totalled or sub–totalled and reports can be generated directly from the card files by the user's specifications.

These infomation management capabilities in addition to the conditional printing capability make this program the system choice for clubs, organizations, or political parties. This combination of decision making logic and text processing is certainly unique in my experience and extremely valuable for any type of public relations or direct membership appeal campaign.

There is, however, a worm in every Apple and there are disadvantages to any system. Specifically, *The Executive Secretary* does not hyphenate, nor does it automatically or semi-automatically handle footnotes and citations at all. Hyphenation is a matter of aesthetics. I have always hated it, and I have always disabled it on any system I used which provided it as an option. If the right and left hand justification is attractive, and it is in this case, then hyphenation is an abomination, at least to me. If it isn't to you, then, you are forwarned. Footnotes are another matter, although they are seldom useful in business reports or letters, they have a firm position in academic writing.

What would you give to have your Apple II able to configure to any peripheral?

SOFTECH MICROSYSTEMS

DISCOUNT DATA PRODUCTS

FOR THE BEST SERVICE
AND PRICE YOUR $ CAN BUY

30% DISCOUNT ‡ MONTHLY SPECIALS

GAMES★ UTILITIES ★ BUSINESS

APPLE		LIST PRICE	OUR PRICE
★ ADVENTURE INTERNATIONAL			
★ APPLIED SOFTWARE			
★ ASTAR INTERNATIONAL COMPANY			
★ ASHTON TATE			
ARTSCI			
Magic Window	D	99.95	79.95
★ AUTOMATED SIMULATIONS			
★ AVALON HILL GAME COMPANY			
AVANT-GARDE CREATIONS			
Hi-Res Secrets	D	125.95	**87.95‡**
Hi-Res Golf	D	29.95	23.95
BEAGLE BROTHERS			
DOS Boss	D	24.00	19.95
Utility City	D	29.50	23.95
Apple Mechanic	D	29.95	23.95
★ BEZ			
BRODERBUND SOFTWARE			
Star Blazer	D	31.95	**25.55**
Space Quarks	D	29.95	23.95
David's Midnight Magic	D	34.95	27.95
Arcade Machine	D	44.95	35.95
Alien Typhoon	D	24.95	19.95
Apple Panic	D	29.95	23.45
Labyrinth	D	29.95	**20.55‡**
Dueling Digits	D	29.95	23.95
BUDGECO			
Raster Blaster	D	29.95	23.95
CALIFORNIA PACIFIC			
3D Graphics	D	39.95	**27.95‡**
Ultima	D	39.95	31.95
CAVALIER COMPUTER			
Aster Field	D	24.95	19.95
Bug Attack	D	29.95	23.95
Micro Wave	D	34.95	**24.45‡**
Star Thief	D	29.95	23.95
★ CONTEXT ★ C.P.U.			
★ CONTINENTAL SOFTWARE			
★ COMPUTER STATION			
DAKIN 5/LEVEL 10			
Programming Aids	D	90.00	71.95
DATAMOST			
County Fair	D	29.95	23.95
Thief	D	29.95	23.95
Casino	D	39.95	31.95
Swashbuckler	D	34.95	**24.55‡**
Snack Attack	D	29.95	23.95
★ DATASOFT ★ DELTA SOFTWARE			
★ DENVER SOFTWARE			
★ EDU-WARE			
GEBELLI SOFTWARE			
Horizon V	D	34.95	27.95
Russki Duck	D	34.95	27.95
HAYDEN SOFTWARE			
Sargon II	D	34.95	27.95
★ HIGHLANDS COMPUTER SERVICES			
★HOWARD SOFTWARE			
INNOVATIVE DESIGN SOFTWARE			
Pool 1.5	D	34.95	27.95
Trick Shot	D	39.95	31.95
Juggler	D	29.95	23.95
INFOCOM			
Zork I	D	39.95	31.95
Zork II	D	39.95	**27.95‡**
Deadline	D	49.95	39.95
I.S.M.			
Mathemagic	D	89.95	71.95
INSOFT			
Graforth II	D	95.00	75.95
★ ISA ★ IUS ★ L&S ★ LINK			
★ LJK ENTERPRISES ★ MICRO LAB			
★ MICRO PRO INTERNATIONAL			
MICROSOFT			
TASC Compiler	D	175.00	139.95
A.L.D.S.	D	125.00	99.95
Olympic Decathlon	D	29.95	23.95
Typing Tutor II	D	24.95	19.95
★ MICROSOFTWARE SYSTEMS ★ M&R			
MUSE			
Castle Wolfenstein	D	29.95	23.95
ON-LINE SYSTEMS			
Ultima II	D	54.95	43.95
Frogger	D	34.95	27.95
#Wizard & Princess	D	32.95	26.35
#3 Cranston Manor	D	34.95	27.95
#4 Ulysess & Golden Fleece	D	39.95	27.95
Screen Writer II	D	129.95	99.95
Expeditor II AppleSoft Com	D	99.95	79.95
Jawbreaker	D	29.95	23.95
Marauder	D	34.95	27.95
Time Zone	D	99.95	79.95
Threshold	D	39.95	31.95
PERSONAL BUSINESS SYSTEM			
Exec. Secretary	D	250.00	**199.95**
PICCADILLY			
Falcons	D	29.95	**20.95‡**
Star Blaster	D	29.95	23.95
Ribbit	D	29.95	23.95
★ PROFESSIONAL SOFTWARE			
★ PROGRAM DESIGN			
★ PHOENIX SOFTWARE			
★ QUALITY SOFTWARE			
★ RIVERBANK SOFTWARE			
SENSIBLE SOFTWARE			
Multi Disk Cata III	D	25.00	19.95
Disk Recovery	D	30.00	23.95
Applesoft-Plus	D	25.00	19.95
Applesoft Pro. Opt	D	20.00	15.95
Disk Org. II	D	30.00	23.95
Quickloader	D	25.00	19.95
Apple Speller	D	75.00	59.95
Image Printer	D	40.00	31.95
SENTIENT SOFTWARE			
Oo-TOPOS	D	32.95	25.95
Congo	D	34.95	27.95
Goldrush	D	34.95	**24.95‡**
★ SIERRA SOFTWARE			
SIRIUS SOFTWARE			
Twerps	D	29.95	**20.95‡**
Snake Byte	D	29.95	23.95
Bandito	D	34.95	27.95
Audex	D	29.95	23.95
Sneakers	D	29.95	23.95
Outpost	D	29.95	23.95
Beer Run	D	29.95	23.95
Borg	D	29.95	23.95
Hadron	D	34.95	27.95
Joyport	A	74.95	**52.95‡**
Gorgon	D	39.95	31.95
Minotaur	D	34.95	27.95
SIR-TECH SOFTWARE			
Wizardy	D	49.95	39.95
Knight of Diamonds	D	34.95	27.95
★ SOFTAPE ★ SOFTWARE EMPORIUM			
SOFTWARE PUBLISHING CORP.			
PFS	D	125.00	99.95
PFS Report	D	95.00	71.95
SOUTHEASTERN			
Data Capture	D	64.95	52.95
Data Capture 80 Cal	D	90.00	69.95
Specify Videx			
Specify Smarterm			
Specify Superterm			
SOUTHWESTERN DATA SYSTEMS			
Ascii Pro	D	129.95	103.95
Doubletime Printer	D	149.95	**104.95‡**
Printographer	D	49.95	39.95
Correspondent	D	59.95	47.95
Merlin	D	64.95	**45.95‡**
The Routine Machine	D	64.95	51.95
Z-Term "The Pro"	D	150.00	119.95
P-Term	D	129.95	**90.95‡**
Norad	D	39.95	31.95
Munch-A-Bug	D	49.95	39.95
STONEWARE PRODUCTS			
D.B. Master	D	229.00	182.95
D.B. Mas. Utly Pak I	D	99.00	79.95
D.B. Mas. Utly Pak 2	D	99.00	**69.30‡**
D.B. Master Stat Pak	D	99.00	**69.30‡**
★ STRATEGIC SIMULATIONS ★ STC			
★ SUBLOGIC ★ SOFTECH			
★ SYNERGISTIC SOFTWARE ★ SORCIM			
TG PRODUCTS			
Game Paddles	A	39.95	31.95
Joystick	A	59.95	47.95
Select A Port	A	59.95	47.95
TURNKEY SOFTWARE			
UNITED SOFTWARE OF AMERICA			
3-D supergraphics	D	39.95	31.95
Super KRAM	D	175.00	139.95
Conquest	D	225.00	**157.50‡**
VISICORP			
Visitrend	D	300.00	229.95
Visicalc	D	250.00	187.50
Visifile	D	250.00	187.50
Visiterm	D	100.00	**69.95‡**
VERSA COMPUTING			
VersaWrit Exp Pac-1	D	39.95	32.95
EZ Port	D	24.95	20.95
Versawrit Expl. Pac II	D	39.95	31.95
★ VIDEX INC. ★ VOYAGER SOFTWARE			

LARGE SELECTION OF ATARI AND TRS 80 SOFTWARE

CALL TOLL FREE 1-800-854-2914 (FOR PLACING ORDERS OUTSIDE CALIFORNIA)

FOR INSIDE CALIFORNIA AND OTHER INQUIRIES CALL 1-714-287-0190

★ SEND FOR OUR FREE PRICE SHEET FOR SPECIFIC PROGRAMS AND PRICES

PLEASE CHECK FOR SYSTEM REQUIREMENTS BEFORE ORDERING. INCLUDE PUBLISHER AND PRODUCT NAME
FOREIGN INQUIRIES INVITED — ADD 10% SHIPPING

All Orders Prepaid by Personal Check, Money Order, Cashier Check. For Faster Delivery Service — Master Charge — Visa

Mastercharge — Visa add 3%, and include expiration date on card.
Please Add $2.50 for Shipping.

California Residents Add 6% Sales Tax
Prices Valid Through This Month's Issue.

Prices Subject to Change Without Notice

DISCOUNT DATA PRODUCTS • P.O. BOX 19674-AF • SAN DIEGO, CALIFORNIA 92119

EVALUATION

This is slowly changing, but may be critical to specifications within the body of the text. What few footnotes you have the courage to present to an editor are included in an appendix so that he can worry about formatting them on his type setting machine. I am fortunate in that they are a problem I will never need to force upon my system, others of you are probably not in that position.

The Executive Secretary was designed to use two disk drives, an 80 column display board with software switching, and a shift key which is implemented in hardware. The no-solder shift key modification provided with the program for newer Apples is the only feature which is absolutely required, various compromises have been implimented for single disk systems. These are compromises.

Every step you take away from the target system increases the awkwardness of the system, but it is possible to grow into your ultimate system slowly. I have a single disk system and 80 column display without the software switching. In fact, my dealer never told me about software switching options. I found the display switching more irritating than the lack of the second disk drive and I plan to remedy that situation almost immediately. This opportunity to move upward in hardware within the same system is a significant advantage over other packages which may have one program for 40 columm display users and another different and additional cost program for 80 column display users.

If you haven't noticed, I am sold. I hope I have made it clear that this program is probably not right for every Apple owner, but if you are one of the individuals that word processors were originally designed for, I can not recommend it enough. There is finally a microcomputer word processing system which should make the commercial vendors nervous. The quality of the documentation and the user interface compares quite favorable to anything on the market today, and the information management capabilities are rare if not absent anywhere else.

What would you give to have TURTLEGRAPHICS, with automatic scaling, and four graphic modes, including HIRES and LORES, on your Apple II?

SOFTECH MICROSYSTEMS

WORD PROCESSOR EVALUATION

The Correspondent

L.K. Bumgarner/reviewer

The Correspondent, evaluated here as a word processor program needs to be considered more correctly as a "text editor." This is the description which its own creators have given it, and certainly one which better fits it. The program is typical of many that have been created for a single purpose and been expanded over time to satisfy the desires of previous purchasers. That is, the program, while quite sophisticated and flexible in the many features it provides, fails to provide the essentials of a true "word processor." In fact, its creators intended the program for the casual user having an occasional letter to write. That function it can perform quite well, and it certainly has use in other text editing fuctions that are not too demanding. Considered in that light, the program is a good value at its current price of $44.95.

The program comes on one disk, supported by a 66 page user manual. Surprisingly, these 66 pages are none too many to describe the many features and controls within the program. In fact this evaluator would like to have seen a bit more description provided in selected areas. This is one of those programs that can be copied up to three times, and as usual it is recommeded that a copy be made to work from and the original disk saved. The new purchaser will soon find that the program instructions are a bit vague, yet force the user to answer a Y/N question to continue before he is advised of the proper position of disks and drives. This produced that instant sinking feeling that responding to the question will blow one of the precious three copies that are allowed. Well, it does not; but in a package of this type, such vagueness should not exist.

Unfortunately the manual makes no mention of what will happen next, once this question is answered and the copy process begins. The disk drive produced no less than six consecutive buzzes which smack of the dreaded I/O error sound, over a period of several seconds. Having already produced doubt in the user's mind as to correct procedure, this symptom is likely to produce the "reset reflex", causing the user to abort the copy prematurely.

Another minor irritant we found was that some new features had been added to the programs after the documentation was printed. These were to be covered in a text file supplied that could be read by the *Correspondent* program and printed out for reference. Unfortunately, we could only view this material on the screen, since attempts to print it out resulted in obtaining about half of the text followed by endless line feeds. A call to the vendor (long distance) produced a suggestion we might try to correct the situation, but it did not work. This leaves the user in the position of having to manually copy pertinent information from the screen. Again, not good testimony to the computer art.

Enough griping about the usual mundane deficiencies we so often find in commercial software these days. Let's get to the essentials of this program as a word processing device.

The program is divided into two primary modes; one for scrolling and one for data entry and editing. This provides the ability to include many more features to the user, but also makes for confusion by the use of similar control codes between the two modes. Also, considerable switching between the two modes is necessary for the user to know where he is within the document being prepared. This becomes frustrating since fundamental movements of the cursor respond to different control codes in the two modes.

Most commands are mnemonic, but still confused by whether they require a preceding control key or not. Most notable of the commands that are not mnemonic are those that govern scrolling, compounded by the fact that they are also different between the two modes. For example, to scroll down in the scroll mode, the left arrow is used. In the edit mode this is achieved by a control "J." Other movements are equally non-similar between the two modes. We also found the cursor to both disappear and lag behind during fast editing moves.

The scroll mode is comprised of no less than 37 separate commands, each accomplished by the press of a single key or a control key and letter. Twenty alphabetic keys are used singly, and fifteen control key sequences are used. This makes it almost impossible for the user to remember whether he wants a "d" or a control "d," unless he uses the program frequently enough to etch the commands into his mind. To compound this problem, some keys are cancelled by a return and some operate on a toggle basis.

The edit mode is similarly designed. Nineteen control key sequences are used, two different escape sequences for capitalization, and three special control codes using dual control key sequences on two of them. All of this provides a lot of capability, but renders the program almost unusable unless a list of all these codes and their functions is kept handy during operation. The authors have wisely included two help menus to aid the user, but it is still necessary to consult the manual for further description in many cases.

As far as speed of the program, I found it to compare very well with the faster word processors so far tested in virtually every category. Search time is remarkably fast, but requires the operator to examine each occurrence and either retype for replacement or carriage return to move to the next qualifying occurence. No global replace is available. One notable deficiency of this approach is that the program always begins the search at the beginning of the text, and gives no provision of page or line numbers when an occurence is found. This necessitates a switch back to the scrolling mode where page and line numbers are provided, with a consequent return to the edit mode in order to achieve the replacement. Nowhere in the edit/data entry mode are line and page number provided. Consequently, when typing in text, one never knows just where they are without switching back and forth to the scroll mode. This is not particularly difficult, but certainly irritating and counter-productive to speed of entry.

EVALUATION

One feature described in the manual as "scroll down one page" actually turns out to be 21 lines. Another reference in the manual says this is 21 lines. If it were, three consecutive keystrokes would bring the top of the next page to view, given that one started at the top of the previous page. Such a feature would have been useful indeed. But at the present scroll of 21 lines, three strokes brings us to line 64 of page 1; succeeding multiples of three keystrokes continue to accumulate this error, requiring the user to arrive at the next page top. Somehow it seems that this should be correctable.

A feature I really liked was the ability to "freeze" up to nine lines at the top of the screen. This feature becomes very useful in forms design and other columnar work, where one may not wish to utilize the tabs that are available in the program. It also allows the operator to compare different sections of text quite readily.

Another nice feature is an optional display mode. When entering text in 80 columns, the screen automatically switches back and forth between the right and left sides as typing takes place. In order to facilitate proofing via the screen, the program allows the operator to select an optional display of 40 columns, thereby cutting each line in half and folding it underneath for easier reading. This works quite well with the exception of an occasional split word.

The procedure described in the manual to reset left margins did not work as indicated. We later found that this was covered in the disk addendum provided, but was in the portion which we were unable to print out. A call to the support organization cleared this up for us.

It is possible with this program to embed control codes within the typed text to perform printer control functions. I found that I was able to activate many of the nifty features of the Epson MX-80 Graftrax with this feature. Some experimentation could prove quite rewarding in this area.

What would you give to develop programs for the IBM PC, TRS 80 Model II, T.I. 99/4 Home Computer, and Xerox 820 on your Apple II?

SOFTECH MICROSYSTEMS

WORD PROCESSOR EVALUATION

Perhaps the most noticeable features missing from the package are the absence of page breaks, generated page numbers, automatic double spacing, and page and line indicators in entry mode. In this respect, *The Correspondent* functions much like a regular typewriter, i.e., the operator must actually type where he wants the page to end and begin, etc..

Such features as block text move and copy are available in this package and work very nicely. It can be used in many ways as a data base manipulator, a letter generator, and can perform such features as columnar addition for checking. All in all it is quite flexible and versatile, but it is not suitable for the professional typist as a word processor. For the computer hobbyist with some text editing requirements who is not overly concerned about speed, and willing to spend some time with the manual on a periodic basis, it should serve fine.

Reply to Bumgarner's review

Roger Wagner/Southwestern Data Systems

Thank you for sending me a copy of the upcoming review of our product "The Correspondent." I appreciate the opportunity to have *Call —A.P.P.L.E.* evaluate our products and inform their membership about the relative merits of the items we offer.

I would like to make a few comments, however. These are offered more as observation and opinions, rather than a specific request that any of the editorial material be changed.

After reading the review, I couldn't help but feel that this particular reviewer felt that the prime purpose of his job was to find any shortcomings of anything reviewed, rather than present a balance of its advantages and disadvantages. I believe the result fails to give your readers what they deserve in both fairness and complete information.

Some cases in point. Mr. Bumgarner mentions that The Correspondent is "one of those programs that can be copied up to three times...." For heavens sake! *The Correspondent* was *The First* program ever to offer the user this option, and S.D.S. as far as I know is the *only* company to use this system for the majority (if any) of their software line! At least a little credit when we do something right, OK?

After spending a considerable amount of editorial space on the shortcomings of the copy program, he then mentions his difficulty in printing out the addendum file. I do apologize for not having located S.D.S. in his home town (so as to avoid the long distance phone call), but his inability to print out the file would seem to be one of "pilot error" in this case, as we have never had anyone else experience any difficulty with the file, and there is nothing complicated in printing out the file that has slowed anyone else down. In

COMPUTER CONSULTING SERVICES

TRUST ACCOUNT MANAGEMENT SYSTEM

1822 DREW STREET, SUITE 3
CLEARWATER, FL. 33515
(813) 446-2643
(813) 441-3490

The people that brought you the Professional Timekeeping System are proud to announce the introduction of another valuable, time-saving office management tool. . . .the C.C.S. Trust Account Management System. Designed with consultation from bankers and legal office managers, this System incorporates all of the principles of basic trust account management into a simple, yet powerful office accounting tool. It expands on these basic principles to give the user instant access to often-requested data such as client trust balances and audit trails for individual client accounts or entire trust accounts.

As an additional feature, C.C.S. has provided a function in this system which will automatically transfer funds from any Trust System account into a client account in the Professional Timekeeping System, making this System a perfect complement to that already powerful legal office management system.

As is our tradition, this system was designed for professionals by professionals. See your local Apple dealer for details.

Hardware Requirements: APPLE II PLUS (48K), DOS 3.3
Dual Disk Drives, Printer

C.C.S. also produces a PROFESSIONAL TIMEKEEPING SYSTEM which simplifies and optimizes the timekeeping and billing process, and provides a professional firm with valuable management information. Call or write us for information.

System Functions:
— Open a new trust account
— Enter deposits and withdrawals
— Print withdrawal checks
— Print account transactions
— Print check/deposit registers
— Void entries
— Analyze an entire trust account
— Analyze a client's account
— Reconcile monthly bank statement
— Purge an existing account
— Directory assistance
— Edit account information
— Change System parameters
— Change System password

Capacities:
One trust account per disk. The System will accommodate any number of accounts. Specific capacities are dependent on the number of client accounts within the trust account.

Typical configuration:
200 clients
1600 accounts entries
8 entries per client, average

(These figures are for active account entries. Records from client accounts with zero balances are periodically removed with the Purge function.)

Would you give $295?

If you're currently using Apple Pascal* on your Apple II, you're probably aware of some noticeable limitations. And you'd probably give a lot for an upgrade package, including the UCSD p-System*, UCSD Pascal* and TURTLEGRAPHICS, that would get your Apple* to do what it's capable of.

Upgrade to the UCSD p-System Version IV from SofTech Microsystems. It's got all the features of Apple Pascal, and then some. For instance, Apple Pascal's UNITS must be linked in at each compilation, the p-System's do not. And instead of being limited to 32 UNITS, like Apple Pascal, the p-System allows a virtually unlimited number.

How about peripheral support? The p-System supports all the peripherals that Apple Pascal does, plus a clock, and a lower case adapter. And, we get more out of the peripherals you've already got—shiftware modification on the keyboard, alpha lock key, typeahead and characters not even on the Apple keyboard.

And when it comes to graphics, our TURTLEGRAPHICS has everything in Apple's graphics, plus automatic scaling and four graphic modes, including both HIRES and LORES.

Then there's portability. The p-System lets you develop genuinely portable, high-level applications for nearly any microcomputer around. It allows you to work in any combination of UCSD Pascal and BASIC (available as an add-on). And it provides support for dynamic memory management and multitasking, with a full arsenal of enhancements. And if that isn't enough, your existing Apple Pascal programs are upward compatible with the p-System, and simply have to be recompiled to execute. All your Apple II needs is 64K of RAM and two disk drives.

Last but not least, there's the price. Normally, you'd have to pay as much as $825 for such a package.

But, for the next two months, we're making this special upgrade offer to Apple Pascal users for a mere $295. That's a savings of over 60%.

So just send in the coupon below, with your proof of purchase and check, money order or Visa or MasterCard number, and you'll be on your way to getting more out of your Apple II than you ever dreamed of. But you'd better hurry. Your two months have already started.

SOFTECH MICROSYSTEMS

Okay, SofTech Microsystems, here's my $295. I want my Apple II to have software it can really appreciate.
My check is enclosed ☐
Please charge to my Acct. #_____
Visa ☐ Master Charge ☐ Expiration Date_____ Name on card_____
I hereby certify that I am an Apple Pascal Owner.
Signature_____
Name_____ **Title**_____
Company_____ **Telephone**_____ **Ext.**_____
Address_____ **City**_____ **State**_____ **Zip**_____

Please send coupon to: **Apple Upgrade Dept.**
P.O. Box 27179, San Diego, CA 92128
Or call (714) 578-6995

My proof of purchase is
☐ invoice ☐ receipt ☐ disk label ☐ other

OFFER VALID JULY 1 to AUGUST 31, 1982

(California residents please add 6% sales tax [California Transit District—6.5%] Massachusetts residents please add 5% sales tax.)

*UCSD p-System and UCSD Pascal are trademarks of the Regents of the University of California. Apple, Apple II, and Apple Pascal are registered trademarks of Apple Computer, Inc.

fact, I remember his specific call, because it was so unusual in the problem he was having. I do think, *especially as a reviewer,* he should have pursued the problem to its solution to determine if it was a software error, or a matter of having forgotten to turn his printer on.

Unfortunately, the rather negative tone persists throughout the article, in spite of his final conclusion that the program in the final analysis will be quite suitable to a number of people. It is sad that towards the end of the article some of the nicer features of the package seem to start occuring to him.

To perhaps even out the information, I would like to offer at least a few of The Correspondent's positive points:

1) User tutorial at beginning of manual allows a computer novice to easily get far enough in the program to enter, save and print a letter.
2) In a time of rapidly increasing software prices, The Correspondent offers a *very* attractive alternative to someone who doesn't want to spend $200–$300 to type a letter.
3) The Correspondent is one of the few products at any price that gives an accurate representation of what the final text will look like, without requiring additional hardware. For producing highly formatted text such as forms (or tests, etc. for teachers) the program has very few competitors – especially at the price!
4) The Correspondent is easily LISTed and modified, and we provide a full-time support staff to help users in any area they require. Although I do not expect a person to have to modify any part of my programs, his own particular preference about how many lines the "page-scroll" should be could have been easily handled while he was calling about his difficulty on printing out the file.
5) The Correspondent is one of a very few programs that automatically configures itself for a variety of user preferences (including printer control and overall program operation), with a minimum of effort from the user.
6) the manual includes a fairly good practice exercise section that goes into the TECHNIQUES of using the varius commands available in the program. Many software products explain the raw commands available in using the package. Few offer advice on how to combine commands to achieve a desired goal.
7) The Correspondent offers a tremendous number of additional utilities and items of interest. These include the ability to send letters to other Apple users directly on diskettes, to put bi-directional scrolling in a user program, along with Correspondent generated text, a form letter driver example, and many others.
8) The fact that compared to Applewriter, The Correspondent is an extremely viable competitor, allowing much more flexibility in terms of the final document created. The Correspondent also gives the user complete flexibility as to whether to save his file as a binary or actual text file format.
9) The ability to link files to unlimited sizes. This is useful for printing documents, but takes on tremendous potential when using The Correspondent as a free-form database (examples provided both on disk and in manual). Combined with a hard-disk, it is possible to search 10 Megabytes of data for a single occurence of a string, located anywhere within the file.

The Correspondent has two main drawbacks. It does not automatically break pages, and words are not automatically moved as whole units to the next line when a character insert or remove is done. (See the accompanying flyer on the package if you are interested) Other than that, it compares quite well against a variety of packages. Even the reviewer's own chart show that.

Well, I certainly have gone on here, and I appreciate your patience in bearing with me through all this. I have very personal feelings about our products, and particularly the ones I have written, but I think I am still reasonably objective in the points I have raised. I look forward to your comments.

Perhaps we could try another.

Sincerely,
Roger Wagner

Benchmark Tests for the A.P.P.L.E. Word Processors Evaluations

	The Correspondent	Executive Secretary
Timing (in seconds)		
Initial boot-up	37	21
Retrieve text	21	10
Store text	26	8
Search	1	2
Search & Replace	Not Auto.	Not Auto.
Printing		
Go to **print** mode from **editor** mode	0	55
Go to **editor** mode from **print** mode	0	55
Print time for test document (Approximately 8,000 characters)	201	415
3 Keystrokes Required:		
Search	1	
Search & Replace	Req. operator each occurrence	
Change a letter	2	

DATA CAPTURE 4.0©

The most advanced and easiest to use telecommunications program for use with the MICROMODEM II™ or the Apple COMMUNICATIONS CARD™.

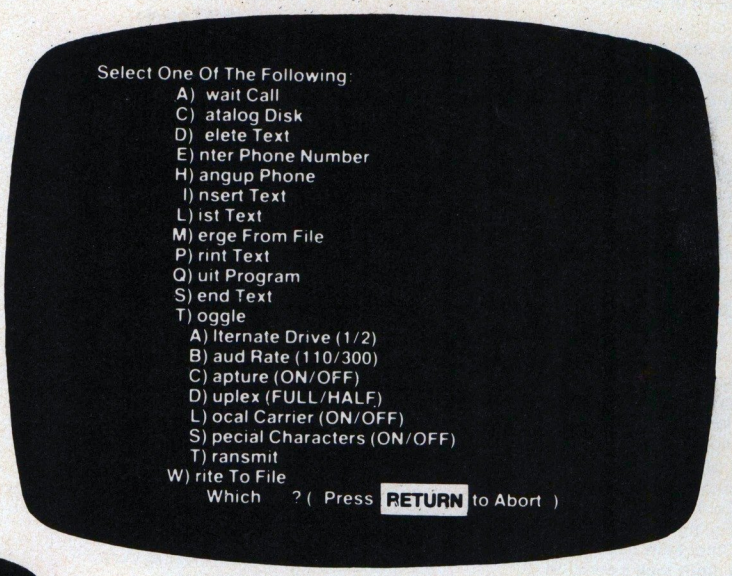
If you use your Apple II™ computer to communicate with other Apples or with timesharing systems, then you need Data Capture 4.0.

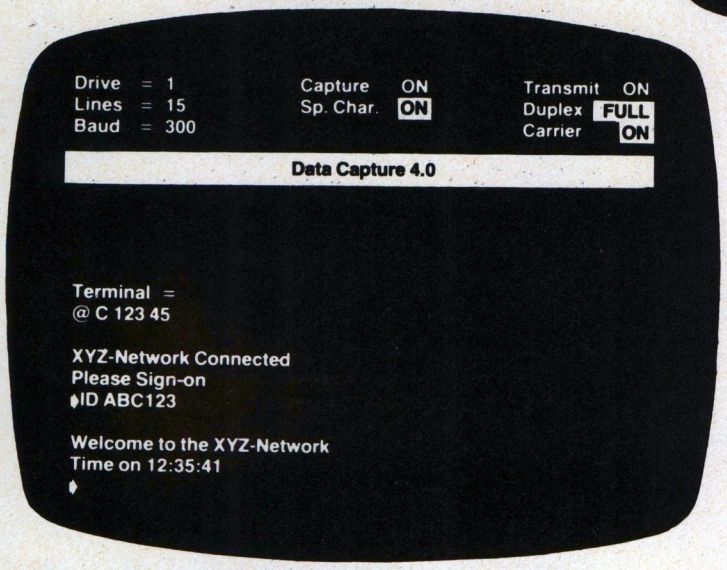
If you want to save stock quotes, airline schedules, electronic mail, or other data from a remote computer system for future use, then you need a smart terminal program for your Apple II™. Data Capture 4.0 is the most user friendly and most flexible program of this type. It is specifically designed to take advantage of all the features of the Micromodem II™, but it works equally well with the Apple Communications Card™ and several other popular serial interface cards.

Data Capture 4.0 is the only Apple II™ smart terminal program available which is fully copyable and modifiable. This means that you don't have to worry about backup. Go ahead and make all the backup copies you need.

The unprotected format and helpful documentation make modifications to Data Capture 4.0 very easy to perform. In addition, full technical support is available by phone from Southeastern Software.

Data Capture 4.0 has many other features. Incoming data files are automatically captured regardless of length. Data in the memory buffer can be viewed, edited, printed, saved to or loaded from disk, or transmitted to the remote system at any time. An unattended mode of operation is provided so that you can call your Apple from another location and send data to it or load data from it. Data Capture 4.0 is fully compatible with the Apple III™ in emulation mode. It is also compatible with all popular lower case adapters for the Apple II™, including the widely used shift key modification. An automatic logon utility for use with the SOURCE is provided with Data Capture 4.0. Also included is a HELP text file containing the latest tips and suggestions on using Data Capture 4.0 effectively (many of these are the result of feedback from customers).





See your local Apple dealer today for a demonstration of Data Capture 4.0 or order direct from Southeastern Software at the address below.

(Please include $2.50 for postage and handling.)

If you presently own a previous version of Data Capture you may upgrade to the new version for the difference in price plus $2.50 for postage and handling. MASTER CARD or VISA orders may be placed by phone.

DATA CAPTURE 4.0 ..... $65.00
DATA CAPTURE 4.0 / 80* $90.00

*Specify either Videx, Smarterm, Double Vision or Sup 'R' Terminal version

— Dealer Inquiries Invited —

Requires DISK II™, Applesoft II™ and 48K of Memory

DATA CAPTURE 4.0© 1980-Southeastern Software

*Apple™, Apple II Plus™, Disk II™ and APPLESOFT II™ are trademarks of Apple Computer Company.

SOURCE – Source Telecomputing Corporation.

*Micromodem™ is a trademark of D.C. Hayes Associates, Inc.

Southeastern Software

6414 Derbyshire Drive ● New Orleans, Louisiana 70126

Dept. CA (504) 246-8438 or (504) 246-7937

master charge THE INTERBANK CARD

BankAmericard VISA

WORD PROCESSOR EVALUATION

	The Correspondent	The Executive Secretary
INITIAL REQUIREMENTS		
Program Cost	$59.95	$250
RAM	48K	48K
Applesoft in ROM	Language Card or X	X
Lower case adapter	Opt.	40 Col. Mode
Number of disk drives	1+	1+
Other hardware modifications		Shift Key
80-column	Opt Partial Use	Opt.
Z-80 Softcard		
RAM Board		
Special Printer		
Special interface card		
TEXT INPUT FEATURES		
1.1 Upper/Lower case display		
Upper case only		
Inverse upper case	X	
Hardware adapter	Opt.	X
High resolution screen		
1.2 Shift key operation		
Escape key sequence	X	
Control key		
Equipment modification	Opt.	Req.
Other keys used		
1.3 Shift lock operation		
Escape key sequence	X	
Control key		CTRL "L"
Equipment modification		
Other keys used		
1.4 Screen display input		
Maximum line length display	40 or 80 w/card	40 or 80 w/card
With scrolling for whole line		
1.7 Automatic formatting of words		
Moves full word to next line	X	X
Automatic indentation		X
First line of new paragraph		X
Temporary for full paragraph	X	
Hyphenation		
Prompt only		
User approves each case	X	
Fully automatic		
1.8 Tabbing		
Available	X	X
Text shows effect during entry	X	
Justification		
Left	X	X
Right	X	X
Center	X	
Decimal Point		
1.9 Status displayed for:		
Current operating mode	X	X
Cursor at column "n"	X	
Cursor at text line "n"	X	
Left margin	X	
Right margin	X	
Top margin		
Bottom margin		
Tabs stops	X	X
Space remaining for more text	X	Automatic File Extension
Other		

	The Correspondent	The Executive Secretary
1.10 Reset page specifications in text		
Margins	X	X
Tabs	X	X
Page breaks		X
1.11 Accept text files it didn't create	X	X
1.12 Accept text from telecomm. port		X
1.14 Formatting for text output		
Center lines	X	X
Underlining	User Embeds CTRL CHRS Used by Printer	X
Bold printing		X
Alternate color ribbon		X
Two-pass printing		X
Super-script/sub-script		X
1.A Automatic checking for spelling		
1.B Maximum text length in one file	?	Full disk in 3,400 word Sub-files
1.C Drops characters from fast typist		
EDITING		
2.1 Underlying document format		
Line oriented editing	X	
Auto adjusts subsequent lines	X	
Document oriented editing		X
Insert text at cursor position	X	X
Delete text from cursor by:		
Character(s)	X	X
Word		X
Sentence		
Paragraph		
Document page		
End of document		
Beginning of document		
Screen line	X	X
Top of screen page		
Bottom of screen page		
Text marker	X	X
Automatic adjustment of text:		
On screen	X	X
In document	X	X
2.2 Cursor positioning — move by:		
2.2.1 Character	X	X
Multiple characters		
Word		X
Sentence		
Paragraph		
Document page		X
End of document	X	X
Beginning of document	X	
Screen line	X	X
Line number		X
Top of screen page		X
Bottom of screen page		X
Scroll forward through text	X	X
Scroll backward through text	X	X
2.2.2 Copy text within file	X	X
Move text within file	X	X
Save portion of text to new file	X	X
Insert text from another file	X	X

	The Correspondent	The Executive Secretary
2.2.3 Search and Replace		
Search Only		
Command finds and displays:		
First occurrence		X
All occurrences	X	
"n"th occurrence		
Search and Replace		
Command replaces:		
First Occurrence		X
All occurrences	X	
"n" occurrences		
Action completed:		
Automatically		
Manual override	X	X
2.4 Operates on multiple file document	X	
2.6 Can edit:		
Page breaks		X
Page number location		X
Headers (page title) location		X
Date		Opt. w/clock
2.7 Columnar editing		
Swap columns		
Addition checking	X	VIP card File
2.8 Are edit macros available for:		
Inserting formatting commands		X
Inserting phrases into the text		X
2.A Indices capabilities		
Automatic creation of indices		X
TEXT STORAGE AND RETRIEVAL		
3.1 Uses		
3.4 Program accessible for modifying		
3.5 Program back-up		
Disk copyable by std. copy util.	Up to 3 copies	
3.6 Text file back-up		
Prompts user		X
Automatic procedure		
Can use std.disk copy utilities	X	
1-drive copy program provided	X	X
2-drive copy program provided		X
3.7 Form letter capability		
Included in basic program	X	X
Extra cost item		
3.8 Text file documentation		
Length of document title	30	15
Content description phrases		X
Allowed		Portion not used for title
Length of phrase		
Indication of document author		
Dating		
Document originated		CCS clock
Document last worked on		CCS clock
Total time document worked on		
Size of the document	X	X
3.9 Amount of text actively in memory	Approx. 18,000	16,000
3.A Does the program support:		
D.I.F. data formats		X
Other data base files	X	X

	The Correspondent	The Executive Secretary
3.B Displays remaining space on disk	X	X
OUTPUT		
4.1 Printer configuration:		
Automatic		
Semi-automatic:choices from list	X	X
Prompts operator in detail		
User changes lines in program	Available	
4.2 Printer control		
Incremental spacing		
Top-of-form alignment	X	
Tabbing	X	
Change print fonts in document:		
Pause for print wheel exchange	X	
Reconfigure dot matrix on fly	X	
4.2.1. Letter space justification	X	
4.2.2 True proportional spacing		
4.4 Forms control		
Continuous form		X
Single sheet		X
Variable length	X	X
4.5 Page numbering		
Suppressed any page		
Continue any page		
Chapter relative		
4.6 Footnotes		
Completely automatic		
Semi-automatic		
4.7 Responds to printer problems		
Out of paper		
Out of ribbon		
Other		
4.8 Print Spooling		
Can print a file while editing another		
4.A Screen preview of final document		
Maximum line length displayed	40 or 80 w/card	84
With scrolling for whole line	X	
Pause key	X	X
Pause at page breaks		X
Escape from viewing entire text		X
Edit page breaks at this time		
4.B Chains files into one document — (automatically)	X	X
HUMAN ENGINEERING		
5.1 Mnemonic quality of commands	Average	Excellent
5.2 Verifies dangerous commands		
Delete document file	X	X
Save file with same name	X	Not Allowed
Quit program		
Clear memory space for new text	X	X
Load file that overwrites memory		Not Allowed
5.3 Recover from command errors		
Unwanted deletion of text		
Unwanted deletion of file		
5.4 Operator assistance		
Operation and error messages		
English	Std. Apple Messages	X
Computer jargon		
Numbers only for table look up		
Help file on-line	X	
5.5 Structure of program		
Separate edit and print programs		
5.6 Emergency recovery provided for:		
System crash — Not power outage	X	
Memory full error		X
DOCUMENTATION AND ASSISTANCE		
6.1 Allows program to be used easily by:		
Average computer hobbyist	X	
Apple user (not hobbyist)		
Average clerical personnel		X
6.2 Identifiable documentation parts		
Intro.-Program purpose & theory	0	+
Tutorial		
Written — using examples	0	0
Computer assisted instruction		
Each function-detail description	0	+
Technical Data		
System requirements		+
Structure of files		
Interfacing with peripherals		
Annotated source program listing		
Full listing		
Part: for user modifications		
Index to documentation	0	+
Quick reference		
Specific section in manual	+	
Pocket reference card		0
Keyboard function chart		
Keyboard key function labels		
On-line screen display	+	
6.6 Program supported by telephone	X	X

NOTES:
Numbers in left-hand column refer to word processor evaluation criteria published in **Call —A.P.P.L.E.**, September, 1980.
A letter suffix indicates an addition since September, 1980.

Key
X = Characteristic present
R = Required
Opt = Optional

\+ = Good
0 = Acceptable
\- = Poor

blank = Characteristic not present or not applicable

/// CORNERS

KEYPAD JOYSTICK

Richard V. Grant

YOUR numeric keypad can be an eight-direction joystick! Any group of keys can become a joystick with enough "IF" statements. The numeric keypad, with its regular pattern of numbers, lends itself to a dramatic short-cut. Consider the following:

```
10  GET A$: A = VAL(A$)
20  UY = INT((A-1)/3)-1:UX = -
    3*UY + A-5
30  X = X + UX:Y = Y + UY
```

Line 10 accepts a keystroke and generates the numeric value "A." Line 20 is the meat of the program, generating unit vectors in the X and Y direction. Line 30 updates the X and Y values controlling point position.

The actual graphic commands are different on the Apple /// and the Apple][and also depend on the intent of the programmer.

In our simple example, we may wish to add the line:

```
15  IF ASC(A$) < 49 OR ASC (A$)
    >57 THEN 10
```

Line 15 makes the routine ignore all keys except 0-9. On the other hand we may wish to insert special options for some letter keys....but that gets into a whole, "You-draw-it," program. Here we just have a short routine which can be used in many programs.

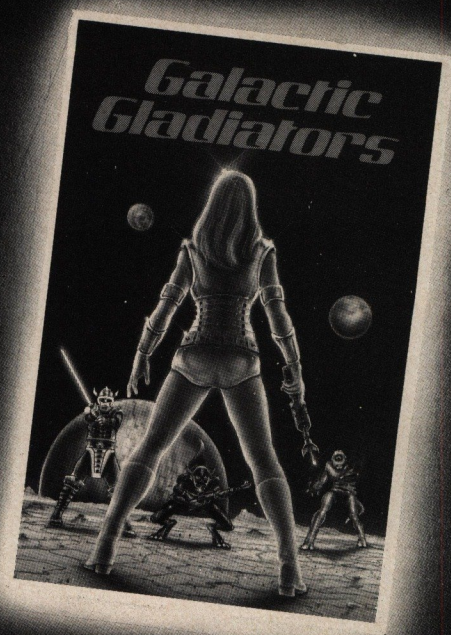
A CALL —A.P.P.L.E. QUICKIE:

PATCH FOR FILEDUMP

William Steinberg

The June 1982 Call —A.P.P.L.E. has a great article on a new DOS command "Filedump". As it now stands, however, the call to print is to COUT1 at $FDF0, the Apple screen. This will cause the text file to be printed to the 40 col screen, even if you have an 80 col board or a printer "hooked". To have this routine print to anything hooked to the DOS I/O, change the patch at byte $BCEB from $F0 to $ED. This will change the print call to COUT at $FDED, and if your printer is on, the file will be printed... If you're using an external terminal, or an 80 col card, or even a modem, the file dump will go there too.

SIZZLING HOT & COMING AT YOU!



RapidFire

GAMES FROM SSI

FUN.

FAST.

POWERFUL.

EXCITING.

LOOK FOR THEM AT YOUR LOCAL COMPUTER/GAME STORE!

If there are no convenient stores near you, VISA and M/C holders can order by calling 800-227-1617, ext. 335 (toll free). In California, call 800-772-3545, ext. 335. To order by mail, send your check to: Strategic Simulations Inc, 465 Fairchild Dr., Suite 108, Mountain View, CA 94043 (California residents, add 6.5% sales tax). Each game is $39.95.

A CALL —A.P.P.L.E. UTILITY:

RESTORE TO A LINE NUMBER ROUTINE

Lee Reynolds/Call —A.P.P.L.E. Staff Writer

FOR quite some time I've been looking for a good application for Applesoft's USR function and at last I've come up with one. The routine described in this article also adds a nice extension to the language, in allowing you to RESTORE to a line number, rather than always to the very first DATA statement in the program.

This capability of choosing which DATA statement to RESTORE to could be quite useful; I'm sure that any creative programmers who might need this sort of thing will be able to think of an application. The general idea behind it is this: suppose you have several categories of things in your DATA statements, and also suppose that you have grouped them by line number in each category. That is, lines between 100 and 200 might contain your first category, lines between 200 and 300 your second category, etc. Now, if you have done this, wouldn't it be convenient to be able to have a statement like RESTORE 200 in order to place the current DATA pointer back to the first item in category 2? Unfortunately, Applesoft doesn't allow this. The following short routine can be inserted in your program in order to be able to set up the USR function to act just like this sort of statement:

```
10  A$="300:20 52 E7 20 1A D6
    90 21 A0 04 B1 9B C9 83 D0
    19 18 A5 9B 69 04 85 7D A5
    9C 69 00 85 7E A5 50 85 7B A5
    51 85 7C A0 00 F0 02 A0 01 A9
    00 20 F2 E2 60 N D9C6G"
20  FOR I=1 TO LEN(A$): POKE
    511+I,ASC(MID$(A$,I,1))+128:
    NEXT: POKE 72,0: CALL -144
30  POKE 11,0:POKE 12,3
```

The first two lines above put the machine language USR routine into memory by use of S.H. Lam's method. The third line sets up the USR vector to point to the start of the routine. The routine has been stored in "page 3" (hex $300 to $330), but it is entirely relocatable, so you could place it somewhere else instead. If you do this, don't forget to modify line 30 above also!

Now, as to how you use it; this is an example:

```
80  X=USR(NUM) : REM RE-
    STORES TO LINE NUMBER
    NUM
```

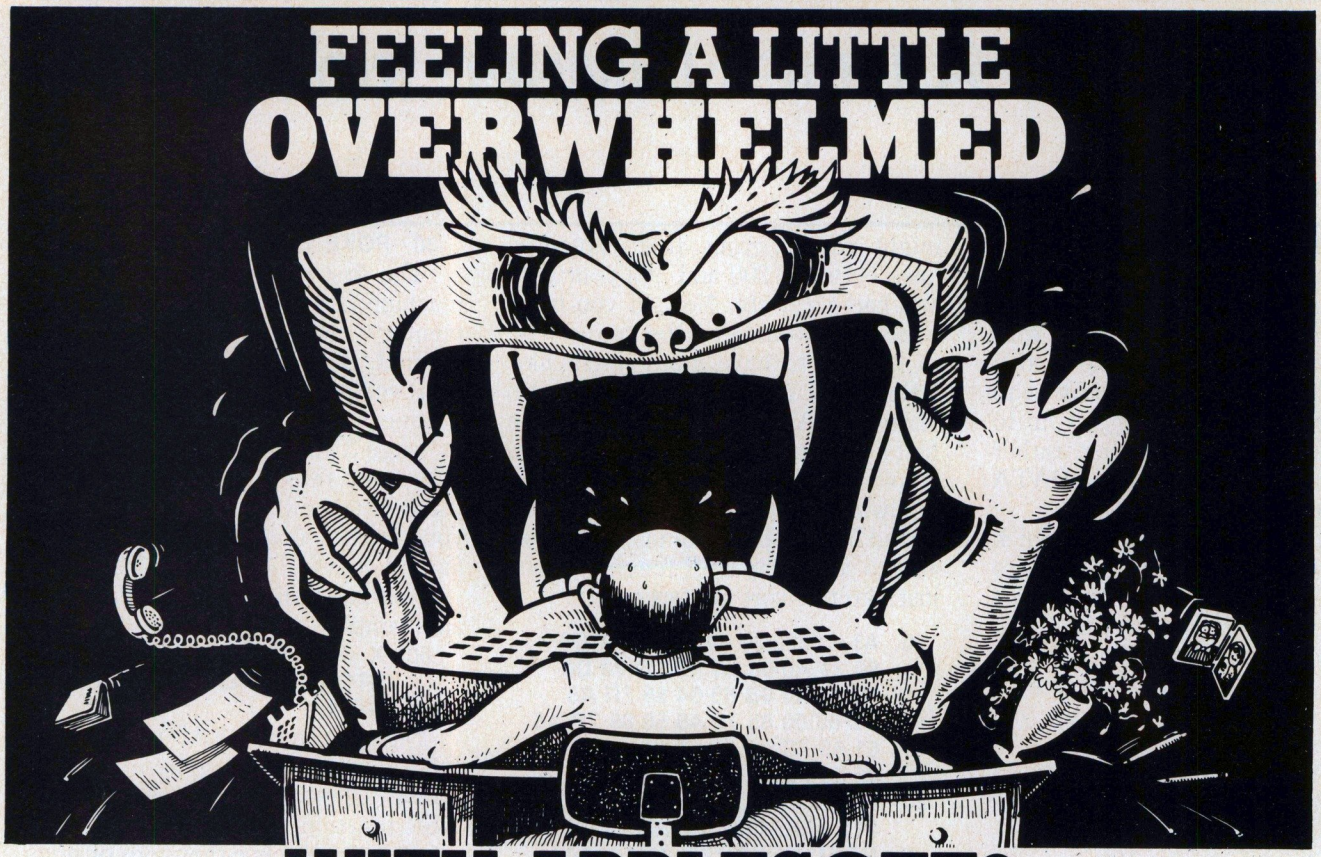
Now your next READ will take its DATA from the first item on line NUM. The value returned by the USR function (and put into X in the above example) is either zero or one. If zero, everything worked O.K. A value of one means an error, which can be one of two kinds: 1) the line NUM isn't in your Applesoft program, or 2) it is, but that line is not a DATA statement! While you're checking your program out, you should probably test the value X.

Here is an assembly listing with comments explaining my routine; it makes use of several routines within the Applesoft interpreter itself:

Anybody got any other neat ideas for using the USR function? If so, I'm sure the readers would like to hear from you.

```
                    ********************************
                    *                              *
                    *   RESTORE to a line number   *
                    *        by  Lee Reynolds      *
                    *                              *
                    * Call -A.P.P.L.E. : July 1982 *
                    *                              *
                    ********************************
                    *
                    LINNUM   EQU  $50
                    DATLIN   EQU  $7B
                    DATPTR   EQU  $7D
                    LOWTR    EQU  $9B
                    FNDLIN   EQU  $D61A
                    GIVAYF   EQU  $E2F2
                    GETADR   EQU  $E752
                    *
                    *
                             ORG  $300
                    *
0300: 20 52 E7               JSR  GETADR     ;Convert USR argument to integer
0303: 20 1A D6               JSR  FNDLIN     ;Find the Applesoft line
0306: 90 21                  BCC  ERR        ;Branch if not found
0308: A0 04                  LDY  #$04       ;Make sure fifth byte in line is the DATA token
030A: B1 9B                  LDA  (LOWTR),Y
030C: C9 83                  CMP  #$83
030E: D0 19                  BNE  ERR        ;Branch if it isn't
0310: 18                     CLC
0311: A5 9B                  LDA  LOWTR
0313: 69 04                  ADC  #$04       ;Point Applesoft's current data
0315: 85 7D                  STA  DATPTR     ;address to the new DATA token
0317: A5 9C                  LDA  LOWTR+1
0319: 69 00                  ADC  #$00
031B: 85 7E                  STA  DATPTR+1
031D: A5 50                  LDA  LINNUM     ;Update Applesoft's value for
031F: 85 7B                  STA  DATLIN     ;current DATA line number
0321: A5 51                  LDA  LINNUM+1
0323: 85 7C                  STA  DATLIN+1
0325: A0 00                  LDY  #$00       ;Low byte (zero) of value to return in "X"
0327: F0 02                  BEQ  OK
0329: A0 01          ERR     LDY  #$01       ;Low byte (one) to signal error
032B: A9 00          OK      LDA  #$00       ;High byte of integer
032D: 20 F2 E2               JSR  GIVAYF     ;Convert from integer to floating point
0330: 60                     RTS             ;Return to Applesoft program
                --End assembly--  49 bytes     Errors: 0
```

FEELING A LITTLE OVERWHELMED



WITH APPLESOFT?

B.E.S.T. Our original Applesoft Optimizer (AOPT) has now been enhanced to bring to the Applesoft programmer the most comprehensive set of software tools available today—B.E.S.T. The BASIC ENHANCED SOFTWARE TOOLS is a machine language program to aid in writing, analyzing, and debugging Applesoft Basic programs. B.E.S.T. will certainly increase programmer efficiency for improved productivity. B.E.S.T. offers: 1) Variable Cross-Reference, 2) Line number Cross-Reference, 3) A Complete Cross-Reference, 4) Optimize variable names with recognition of both short (two character/ Applesoft standard) and long labels, 5) REM Optimizer—with an option to "protect" specified REM statements, 6) Line Optimizer to join short instruction lines of specified length, 7) Merging two programs, 8) Automatic line numbering, and 9) A powerful, yet easy to use Renumber function. Implementation of B.E.S.T. can reduce memory requirements by up to 50% and increase execution speed of Applesoft programs by up to 100%. B.E.S.T. is a low cost alternative to an Applesoft compiler.

B.E.S.T. requires a 48K Apple II/II+, Applesoft in ROM or Language card, and DOS 3.3..... **$40.00**

EDIT-SOFT. A powerful, yet affordable, line editor for Applesoft. Using EDIT-SOFT, you can drastically cut your programming time. EDIT-SOFT not only contains the standard line editor features like inserting or deleting characters, moving to a specific character, entry of lower case letters, going to the beginning or end of a line, and displaying control characters, but it also has the advanced features that will prove indispensable:

- AUTO LINE NUMBERING
- SPLICING two lines together
- AUTOMATIC CHARACTER COUNTER for quoted strings makes screen formatting a snap
- EDITING LINES as they are being TYPED
- A STATUS LINE keeps you constantly aware of which options are currently being used
- Up to TWENTY MACROS available at any time
- MACRO EDITING AT ANY TIME
- HELP PAGES available when needed

Compare the features of EDIT-SOFT to other line editors, then compare the price. No other line editor has so many features at such a reasonable price!

EDIT-SOFT requires 48K of RAM, Applesoft in ROM (language and RAM expansion cards are fine), and DOS 3.3. ONLY **$30.00**

APLUS. The original Structured Basic enhancement for Applesoft Basic. APLUS provides the tools that will increase your efficiency, simplify program rewriting, and make your program flow more understandable. No longer do you have to remember the line numbers of subroutines and procedures; instead, just "DO GRAPHIT" and APLUS does the rest. APLUS adds the following structured programming commands to Applesoft basic: WHEN..ELSE..FIN, UNTIL, WHILE, CASE, SELECT, and OTHERWISE. Multi-line IF..FIN statements are also supported. APLUS includes functions to output indented listings to clarify logic flow and converting source code to standard Applesoft commands.

Requires Applesoft, 32K of RAM, and DOS 3.3...Only **$25.00.**

BUILD USING. Do you have trouble printing charts, reports, or formatting numbers on the screen? If you do, BUILD USING can solve your problems. BUILD USING is a powerful Applesoft utility which provides a "print-using" type routine for numbers and strings. By creating simple "formats", you tell BUILD USING how to format the output. The output from BUILD USING are strings which may be printed, written to disk, saved for later usage, or even reformatted. With BUILD USING, you can choose how many digits should be displayed to the left and right of the decimal point, and even fill the leading positions with the character of your choice. For example, you can print the number '157.23' and '157.2' or '000157.230', of '****$157. AND 23/100 DOLLARS', or hundreds or other ways (including exponential formats). Working with strings is just as easy. Also included are three levels of error trapping, so you can correct numbers that cannot fit into your specified format.

Utilities like BUILD USING are usually difficult to use because they must be located in one memory location (usually between DOS and the DOS file buffers), they cannot be used with your favorite editor or other special routines. BUILD USING does not have this limitation, as it can be easily located in many different memory locations: 1) the "standard" between DOS and DOS file buffers, 2) at HIMEM, 3) APPENDED to your Applesoft program, or 4) anywhere else in memory. Appending BUILD USING to your program is as simple as EXECing a TEXT file. BUILD USING uses the "CALL" command thereby leaving the ampersand vector free for your own use.

BUILD USING requires Applesoft in ROM (Language cards are fine), DOS 3.3 and a minimum of 32K...Only **$30.00**

Sensible Software
6619 Perham Drive Dept. C
West Bloomfield, Michigan 48033
(313) 399-8877

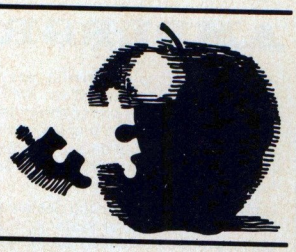
Please specify program desired.
Visa and Mastercard Welcome. Add $1.25 postage and handling per diskette.

• Applesoft is a registered trademark of APPLE Computer Company •

SO WHAT DID YOU EXPECT?

UNRAVELING AN APPLESOFT PUZZLE

Cornelis Bongers/Call —A.P.P.L.E. Staff Writer



Introduction

EVERY computer user is confronted with confusing situations regularly. In particular, people who just start programming, frequently encounter problems not mentioned in any manual. The reaction of almost every beginner when their computer refuses to execute their instructions, or worse, when it wanders off in never-never land for the first time, is that there must be a system failure somewhere. I vividly remember running memory tests almost daily during the first months I had my Apple, as I was convinced that a hidden bug in the hardware had to be the cause of my numerous system crashes. However, time and again it appeared that the crash was caused by programming errors. Instead of testing memory, I began looking more critically to my own efforts. This worked alright till, about two months ago, I spent three days searching a mysterious bug in a complex machine language sort program. Finally, it occurred to me that a hardware failure was not completely to be excluded, so I got the (dusty) diskette with the memory test. On execution, the test reported — more or less to my surprise — a RAM failure. After having replaced the bad RAM, the program worked fine. Although the odds of a hardware failure versus a program error are very low, it may be worth the trouble to check your system now and then.

When programming in BASIC, a third potential *crash causing* factor must be taken into account, namely Applesoft. Applesoft has some internal bugs, most of which are well documented by now. However, there still are *fortunately* some opportunities to discover new mysteries. One of these was pointed out to me by Val Golding (who saw it in *Softalk*) and it occurs in the following program.

```
5   INPUT A$:RETURN
100 TEXT:HOME:VTAB 10:PRINT
    "APPLESOFT ERROR"
```

Enter the program exactly as shown and next, enter the *direct* command GOSUB5 (don't put a blank between "GOSUB" and "5"). On the input request, type ERR. Contrary to what you may expect, the program will print the text "APPLESOFT ERROR," although line 100 is not executed at all. To show how stubborn the Apple can be, delete line 100 by entering 100. Next, execute the GOSUB5 instruction again and you see that the text "APPLESOFT ERROR" will still be displayed.

The cause of the strange behavior of the Apple is neither a hardware problem nor a bug in the program, but rather the occurrence of a situation not foreseen by the creators of Applesoft. The error is caused by the mixture of immediate and deferred mode instructions that are being executed. Furthermore, the error occurs not only in Applesoft, but also on the CBM 4032, as was checked out by my colleague Hans Geilenkirchen.

Explanation

When entering a sequence of immediate mode instructions, these are tokenized and stored in the input buffer. Then Applesoft executes the instructions (just like a normal program line), but it sets a pointer to indicate that the execution mode is immediate. The pointer corresponds to the high byte of the current line number: this byte is set to $FF if the mode is immediate. As is well known, not all BASIC commands are allowed in immediate execution mode. For instance, the INPUT and GET statements are forbidden. The reason is that by specifying INPUT A$ in immediate execution mode, the input buffer would be used for two purposes simultaneously. The users input on the INPUT statement is namely also stored in the input buffer, which would have the consequence that the user's input overwrites the sequence of instructions (that were previously stored in the input buffer).

Since the GOSUB statement is allowed in immediate execution mode, we have the opportunity to trick Applesoft by letting it think it is running a program, whereas in fact the execution mode is immediate. On executing the GOSUB statement, Applesoft's GOSUB handler saves on the stack (among others) the text-pointer, which points at that time to the first digit of the line number following the GOSUB. Next, the GOSUB handler transfers control to the GOTO handler, which subsequently converts the line number to hex, searches for the line number in the program, saves it in the current line number location and starts execution of the specified line.

Note that now the high byte of the current line number does not contain $FF anymore, so Applesoft will from now on think it is running a program. Therefore, Applesoft doesn't generate an ILLEGAL DIRECT ERROR on encountering the INPUT A$ statement (which would be the right thing to do), but executes the statement instead.

BYTES & PIECES

WOW! WHAT SUMMER SPECIALS

Apple Software • All Programs On Disk

	List	Special
Apple Panic	29.95	22.15
Arcade Machine	44.95	33.16
Borg	29.95	22.15
Castle Wolfenstein	29.95	22.15
David's Midnite Magic	34.95	26.76
Dung Beetle	29.95	22.15
Hadron	34.95	26.76
Kabul Spy	34.95	26.76
Star Blazer	31.95	24.16
Sub-Logic Pinball	29.95	23.96
Track Attack	29.95	22.15
Warp Destroyer	29.95	22.15
Wizardry #2	34.95	26.76
Zork I or II	39.95	29.85
Alpha Plot	39.50	29.45
Apple Speller	75.00	57.96
D.B. Master	229.00	173.96
Data Capture 4.0	64.95	49.85
Executive Secretary	250.00	188.86
Home Accountant	74.95	57.96
Screenwriter II	129.95	99.96
Visicalc 3.3	250.00	201.16
Visitrend/Visiplot	300.00	241.96
Zoom Grafix	39.95	29.85

C.O.D. • Money Orders • Certified Checks • Personal Check Allow 2 Weeks
N.Y.S. Res. Add 7.25% Sales Tax • Orders Under $100. Add P&H
$2.00 U.S., $2.50 Canada, U.S. Funds
Foreign Charges Only Min. P&H $6.00

• SEND FOR FREE PRICE LIST #267 •

BYTES & PIECES (516) 751-2535

Box 525 Dept. G6 • E. Setauket, N.Y. 11733

AMPERGRAPH

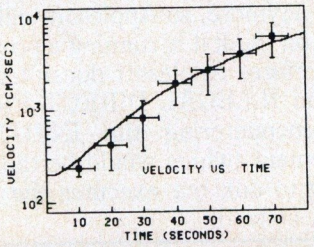
AMPERGRAPH is a powerful, easy-to-use graphics utility for the Apple II Plus. AMPERGRAPH adds twenty-two Applesoft commands that allow effortless generation of professional-looking plots of scientific or financial data. All of the necessary scaling and screen formatting is accomplished with just a few, simple Applesoft lines.

Unlike most other plotting systems for the Apple II which are stand-alone systems, the AMPERGRAPH utility provides extended BASIC graphics language macros that you can use directly in your own Applesoft programs. The additional commands are &SCALE, &LIMIT, &AXES, &GRID, &FRAME, &LOG X, &LOG Y, &LABEL AXES, &LABEL, &VLABEL, &CENTER LABEL, &CENTER VLABEL, &DRAW, &PENUP, &CROSS, &OPEN SQUARE, &CLOSED SQUARE, &OPEN CIRCLE, &CLOSED CIRCLE, &ERROR BARS, &DUMP (to dump the graph on a Silentype printer) and &*DUMP (to link with AMPERDUMP, see below).

AMPERGRAPH uses the Applesoft ampersand machine language jump vector to link to a relocatable 9K routine which normally resides above the second page of high-resolution graphics in the Apple II Plus.

SAMPLE AMPERGRAPH PROGRAM LISTING:

```
10 &SCALE, 0, 80, 80, 13000
15 LX$ = "TIME (SECONDS)":LY$ = "VELOCITY
   (CM/SEC)"
20 &LOG Y:&LABEL AXES, 10, 10
25 LABEL$ = "VELOCITY VS. TIME":&LABEL, 30,
   200
30 FOR T = 0 TO 80:&DRAW, T, 150 + T12:NEXT T
35 FOR T = 10 TO 70 STEP 10
40 &CLOSED SQUARE, T,
   (150 + T12)*(.8 + .4*RND(3) )
45 &ERROR BARS, 5, T12/2
50 NEXT T:&DUMP
```



AMPERDUMP

AMPERDUMP is a high-resolution graphics dump utility which was written specifically to take advantage of the graphics features of the Epson MX-80 and MX-100 printers (MX-80 must have the Graftrax conversion). AMPERDUMP offers many features which are not available in other graphics dump routines:

- Three horizontal magnifications (2.33, 4.66 and 6.99 inches wide)
- Nine vertical magnifications with the MX-80 (0.88, 1.77, 2.64, 3.78, 4.25, 4.45, 5.31, 5.87, and 7.96 inches high); and three vertical magnifications with the MX-100 (2.64, 5.31, and 7.96 inches high)
- Horizontal and vertical magnifications can be specified independently to produce 27 different plot size formats with the MX-80, and 9 different formats with the MX-100
- Normal / Inverse dumps
- Fast
- Adjustable horizontal tab
- Easy to use
- Compatible with AMPERGRAPH
- Relocatable

The AMPERGRAPH and AMPERDUMP graphics utilities require and Apple II Plus (or Apple II with language card) with 48K and DOS 3.3. The AMPERDUMP utility requires and Epson MX-80 with Graftrax, or an MX-100, and one of the following interface cards: Epson, Apple, Grappler, Interactive Structures, or Mountain Computer.

AMPERGRAPH and AMPERDUMP are available from your dealer for $30.00 each, or order direct. Include $1.50 for shipping and handling; Wisconsin residents add 4% sales tax.

madWest SOFTWARE

DEP'T C P.O. Box 9822

Madison, WI 53715

608-238-4875

UNRAVELING A PUZZLE

It will be clear that by entering a long string, problems will arise, for sooner or later, Applesoft will encounter a RETURN statement which transfers control back to the instructions that are stored in the input buffer. As Applesoft will find there the inputted character string rather than the tokenized BASIC instructions, a crash will usually occur.

The reason the program above doesn't crash is that Applesoft is tricked once again, whereby use is made of the fact that Applesoft stores a zero in the input buffer *at the end of each input line.*

If program line 100 has been entered, the exact contents of the input buffer are as displayed in column 2 of figure 1. As can be seen, Applesoft has stored a zero at $21A and $21C. These zeros indicate that execution must be terminated since the end of the instruction sequence is reached.

When executing GOSUB5[return], Applesoft saves on the stack the value $201, being the current value of the textpointer. Next, the INPUT A$ statement is executed and the result is that the inputted text ERR is stored at $200 and a zero, indicating the end of text at $203 (see column 4). Then Applesoft encounters the RETURN, and so pulls Textpointer from the stack again. The RETURN handler searches for the next end of statement symbol which is a colon or a (hex) zero. (Note that the RETURN handler doesn't check the characters following the GOSUB token, so statements like: 100 GOSUB 1000 without a colon or 100 GOSUB 1000/3 will also be accepted and executed by Applesoft).

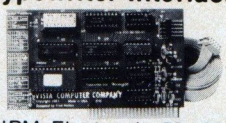
Because the restored textpointer equals $201 and location $203 contains a zero, Applesoft will think that the end of the line is at $203. To check whether the program is finished, Applesoft compares the byte stored in $205 with zero. Since $205 contains the value 53, execution is continued. However, as Applesoft still thinks it is running a program, it will treat the bytes at $204 and $205 as the pointer to the next line and the bytes at $206 and $207 as the (new) current line number. The instruction starting at $208 is then executed. Since this instruction corresponds to the last part of line 100 previously entered and still present in the input buffer, Applesoft kindly prints "APPLESOFT ERROR" before control is returned to the user.

Figure 1 Contents of the input buffer after the different inputs

input buffer location	(1) "after" line 5	(2) "after" line 100	(3) "after" GOSUB5	(4) "after" INPUT A$
$200	INPUT	TEXT	GOSUB	E
$201	A	:	5	R
$202	$	HOME	0 (EOL)	R
$203	:	:	85 (U)	0 (EOT)
$204	RETURN	VTAB	0 (EOP)	0
$205	0 (EOL)	1	53 (5)	53 (5)
$206	—	0	0 (EOT)	0
$207	0 (EOP)	:	:	:
$208	—	PRINT	PRINT	PRINT
$209	—	"	"	"
$20A	—	<-- APPLESOFT ERROR-->		
$219	—	"	"	"
$21A	—	0 (EOL)	0 (EOL)	0 (EOL)
$21B	—	—	—	—
$21C	—	0 (EOP)	0 (EOP)	0 (EOP)

Explanation:
EOT: End Of Text
EOL: End Of Line
EOP: End Of Program

MODEL GB75©
Typewriter Interface

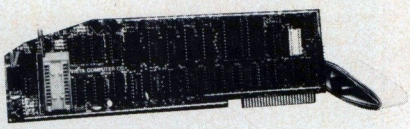


Apple to IBM Electronic 50, 60, 75 Typewriters Interface

- Reads IBM keyboard in parallel with Apple keyboard
- Supports the IBM code functions using an escape sequence
- Types at about 13 characters per second
- Prints from Integer or Applesoft programs
- Supports the "Control I Number N" parallel line length mode sequence
- Has switch selectable upper/lower case I/O. 60, 66, 78 continuous form feed page lengths, 40+video, 80, 95, 132 character line lengths

Suggested price **$225.00**

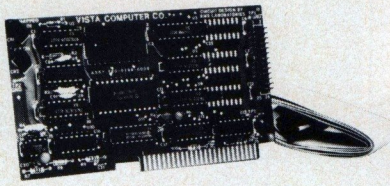
PROM DEVELOPMENT SYSTEM©



- Menu driven program development monitor
- Programs 2708, 2716, 2532, 2732 and 48016 EPROMS
- Simulates PROM from RAM (4K)
- Data and address interface for operator location and control
- Complete user documentation

Suggested price **$495.00**

VISION 40©

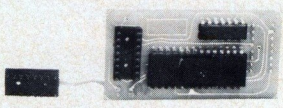


Softscreen programmable character/generator card for the Apple II computer

- Allows use of DOS tool kit upper/lower case character sets in Apple 40 column mode
- Permits creation of new alpha/numeric and graphic characters under Aminatrix
- Ideal for non-English language applications
- Compatible with most popular word processing software packages

Suggested price **$195.00**

MODEL 150©
Type Ahead Buffer



- Up to 40 character type ahead capability
- Enter commands or data while your Apple is processing previous instructions
- Compatible with all Apple computers, keyboards and software
- No cuts — no jumpers — no software patches required
- Includes complete instructions for quick and easy installation

Suggested price **$49.95**

you have one apple.™ now get a whole orchard

of apple add-ons from

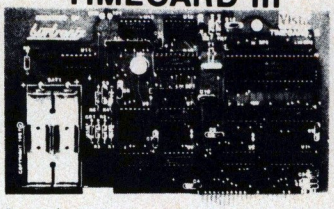
Vista COMPUTER COMPANY, inc.

1317 East Edinger
Santa Ana, CA 92705
714-953-0523

Available through your local computer dealer.

© Copyright 1981 Vista Computer Company, Inc
™Apple Computer Company, Inc
™Digital Research, Inc. © Designed by Burtronix

TIMECARD III©

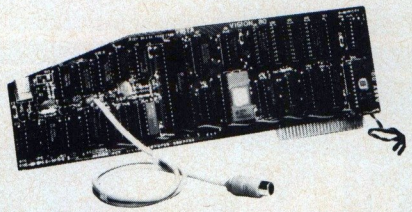


Multi-function time utility for the APPLE III computer system. Contains the year of the century, the month, the date, the day of week, the hour, the minute, the second.

- A countdown timer with a range of one millisecond to 999 hours, 59 minutes, 59 second, 999 milliseconds
- Selectable 12 or 24 hour time formats
- Diagnostic error reporting
- Fully compatible with the APPLE SOS operating system

Suggested price **$195.00**

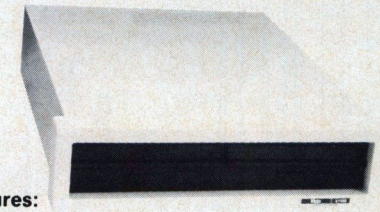
VISION 80©
Video Display Card



- Full upper and lower case character capability with 3 dot descenders
- 9x10 dot matrix per line U.S. (9x11 Europe)
- 128 ASCII character set
- BASIC, FORTRAN and Pascal languages supported
- Z80™ and CP/M™ compatible
- Compatible with all standard Apple™ peripherals
- Shift and lock for upper and lower case
- Source switches between 40x24 and 80x24 software and hardware
- Rated #1 video card by **Softalk** and **Call Apple**

Suggested price **$395.00**

V-1000



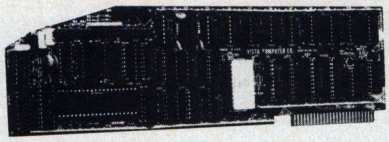
Features:

- Deluxe chassis with internal slide allows easy access
- Storage capacity from 250 kilobytes to 2.5 megabytes
- Accommodates both single-sided and double-sided drives

Prices:

Cabinet with (2) single-sided drives w/power supply $1795.00
Cabinet with (2) double-sided drives w/power supply $2495.00
Cabinet w/power supply ..... $ 495.00

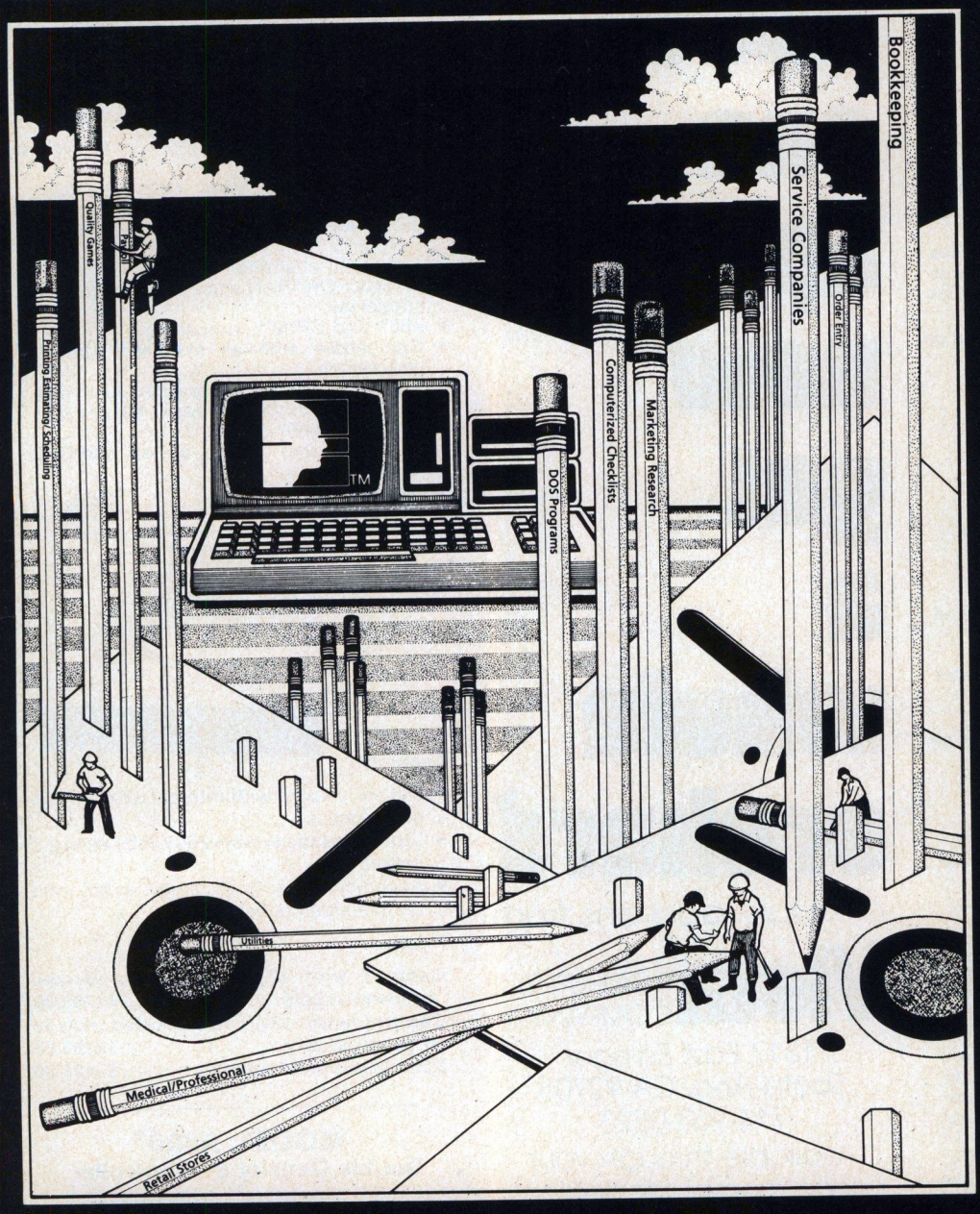
MODEL A800©
Double Density 8" Controller



- High speed DMA transfer of data (1 micro-second/byte)
- Documentation provided — includes theory of operation and diskettes
- Uses all standard Apple DOS commands (OPEN, CATALOG, LOCK, DELETE, LOAD, ETC.) except for INIT which has been improved and enhanced in a Vista format routine
- Compatible with Apple DOS 3.3, Pascal 1.1 and CPM 2.2 (with the Z80 soft card by Microsoft)
- 2K x 8 PROM contains Autoboot functions and all eight-inch floppy driver code allowing complete compatibility with Apple DOS 3.3

Suggested price **$595.00**

WHEN IT COMES TO SELLING SOFTWARE . . . WE GET TO THE POINT.



JOIN THE SOFTWARE AUTHORITY

Writing computer programs is what you're good at. But no matter how good your programs are, they won't sell themselves. Someone has to take the time and effort to market your works. To make them sell. Selling . . . that's what we're good at.

At Advanced Operating Systems, we're looking for high-quality software. We'll take your finished programs and relieve you of the hassles involved with publishing and distributing, giving you more time to sharpen your skills as a programmer.

We're proud of our team of authors and if you have what it takes, we'd like to add your name to our team. When you team up with Advanced Operating Systems, you can expect professional yet personal attention from our expert staff.

Royalties have never been treated so well. If your program is accepted, we provide highly-competitive royalties and an unequaled marketing effort. In addition, we offer you the security of dealing with a company that will reliably send your royalty checks.

As a division of Howard W. Sams, a wholly-owned subsidiary of ITT, we carry a reputation for excellence in technical service. Let us put this reputation to work for you today. Our 150 representatives in the field will sell your product both to computer stores and directly to the consumer. Your program will benefit from attractive package design, effective copy, and professional layout design.

TYPES OF PROGRAMS WE'RE LOOKING FOR

Applications for:
Retail Stores
Service Companies
Medical/Professional
Quality Games
Bookkeeping
Order Entry
Payroll
DOS Programs
Computerized Checklists
Marketing Research
Printing Estimating/Scheduling

WE ARE CURRENTLY REVIEWING PROGRAMS COMPATIBLE WITH:

Apple
IBM
TRS-80
CP/M
Atari
Commodore
Sinclair
Osborne

■ Contact us and we'll send you an informative brochure, along with a questionnaire and non-disclosure form. Call or write:

Palmer T. Wolf

ADVANCED OPERATING SYSTEMS
450 St. John Road
Michigan City, IN 46360
(800) 348-8558 In Indiana Call:
(219) 879-4693

PATCHING DOS THE EASY WAY

Leighton Paul

MANY Apple users like to customize their DOS a little bit, adding patches published in *Call —A.P.P.L.E.* and other sources. For example, patches to DOS have been published to allow direct keyboard typing of the textfile commands OPEN, WRITE, and CLOSE. Other patches prevent automatic reloading of the language card on bootup, and things like making the greeting program a binary file or EXEC file.

Making a patch to DOS that is permanent has usually involved several steps:

Because changing the image of DOS that exists in RAM memory is effective only until the next bootup, this means that the image of DOS stored on tracks 0–2 of the diskette must be changed. The first step involved finding the place in RAM memory where the patch was to reside.

The next step involved using a disk utility program to read individual tracks and sectors of tracks 0 – 2 on the diskette to find where the same code resides on the disk.

The final steps were to use the disk utility program to change the necessary bytes on the disk, then boot up with it and try the patches out.

The following short program is a utility to greatly speed the patching process. It also allows the user to change the name of the program that is automatically run on bootup. Written for a 48K system, it uses the DOS file manager to store the RAM image of DOS back to the disk, the same way that the INIT command does, but without reformatting the disk, or erasing any files.

To make patches to DOS on a disk, all the user needs to do is LOAD DOS UPDATE, make the desired patches in the RAM image of DOS, then type RUN from BASIC.

Here is an example of how to patch DOS to avoid reloading the language card on bootup:

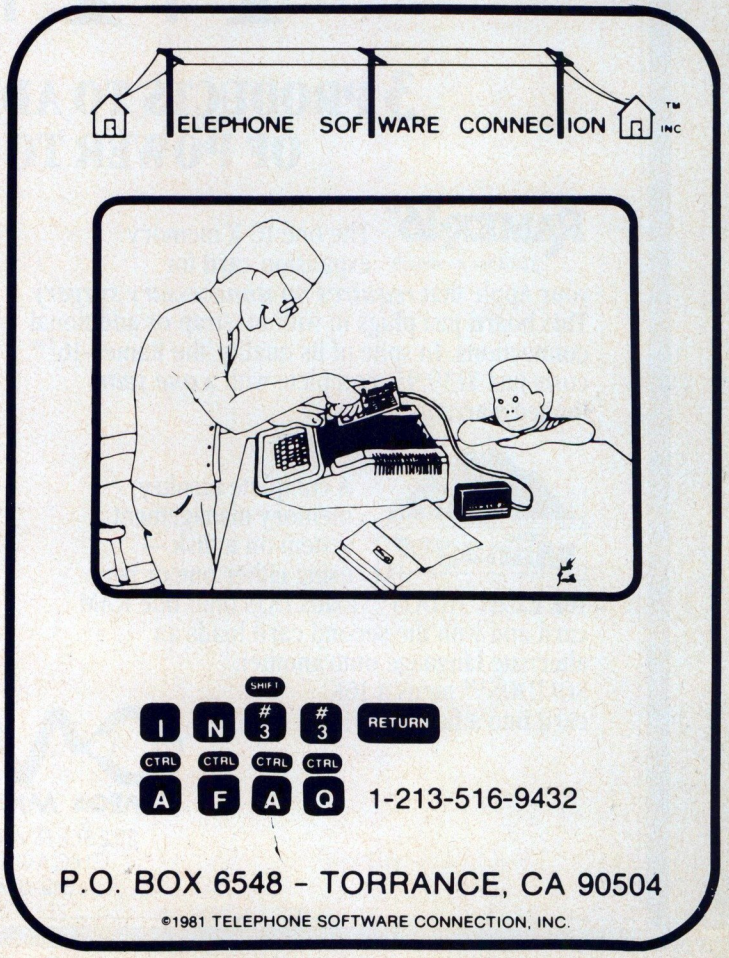
```
]LOAD DOS UPDATE
]CALL-151
*BFD3:EA EA EA
*3D0G
]RUN
```

After typing RUN, the program will ask for the name of the greeting program to be stored in the DOS image. If it is HELLO, just type HELLO, and hit return. The DOS on the disk will then be updated.

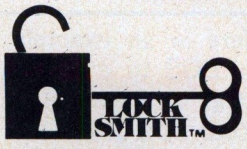
The program should be run *only* on standard DOS 3.3 disks originally formatted with the INIT command. It should never be used on a copy-protected disk containing non-standard DOS.

This same program is also available, already typed and tested, for FREE by calling Telephone Software Connection's 24–hour MODEM number (213) 516–9432.

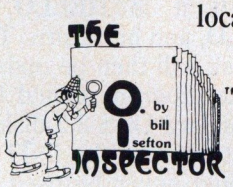
```
]LIST
10  REM
 DOS UPDATE  BY LEIGHTON PAUL
CALL  - A.P.P.L.E. : JULY 1982
(48K APPLE ][ ONLY)

40   FOR I = 32768 TO 32822: READ
      A: POKE I,A: NEXT
50   PRINT "ENTER THE NAME OF THE
      GREETING PROGRAM"
55   PRINT "TO BE RUN AUTOMATICALL
      Y UPON BOOTING UP:"
60   PRINT : INPUT "FILENAME: ";F$

70   IF  NOT  LEN (F$) THEN 60
80 T =  ASC ( LEFT$ (F$,1)): IF T
      < 64 OR T > 90 THEN  PRINT
     : PRINT "ILLEGAL FILENAME!":
      GOTO 60
90   POKE 32792, LEN (F$) - 1
100   FOR I = 32823 TO 32823 +  LEN
      (F$) - 1: POKE I, ASC ( MID$
      (F$,I - 32822,1)) + 128: NEXT
     : REM  STORE FILENAME
110   CALL 32768: REM   STORE DOS
      ON DISK
2000   DATA 186,142,155,179,162,15
      ,189,37,128,157,187,181,202,
      16,247,169,0,141,95,170,32,1
      49,160,160,4,185,55,128,153,
      117,170,136,16,247,76,255,17
      4,11,157,1,0,254,1,15,0,117,
      170,0,0,0,152,0,151,0,150
```

TOGETHER, LOCKSMITH™, THE INSPECTOR™ AND WATSON™ GIVE YOU TOTAL CONTROL OF YOUR APPLE AND ITS DISKS.



Our new 4.1 version is by far the most reliable nibble-copy program for the Apple. There simply *is* no competition. Allows you to backup just about *any* diskette. Includes read/write Nibble Editor, Quickscan Analysis, Media Surface Check, Degauss and Erase, Inspector Interface and Disk-drive Speed Calibration utilities. All for just $99.95 at your local dealer or direct.



Puts all your disk and memory utilities together where they belong — *inside* your Apple. Eprom or disk version is always at your fingertips. Search memory and disks forward and backwards, read nibbles, map disk space, locate strings, the uses are endless. At your local dealer or direct— THE INSPECTOR, $59.95.

Watson™ T.M. The Inspector's Assistant

Includes such goodies as scrolling screen dump, disassembler that shows ASCII, file follower of file, track/sector list-finder by *name*, disk-sector lockout, disk comparer, much more. At your local dealer or direct— WATSON,* $49.95.

If you're at all serious about programming or about business use of your Apple, you must have these interactive utilities. MasterCard and Visa holders order toll-free, 1-800-835-2246.

OMEGA MICROWARE, INC. T.M.

222 SO. RIVERSIDE PLAZA • CHICAGO, IL 60606 • 312-648-4844

*Requires The Inspector

Apple is a registered trademark of Apple Computer, Inc.

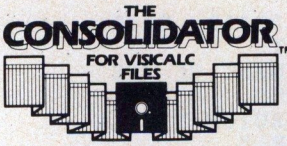
1+1+1=4

3 PRODUCTS TO ADD A 4TH DIMENSION OF POWER TO YOUR APPLE.

RAMEX-16™ *(Look Ma, no straps!)*

The one 16-K memory expansion card for your Apple that *requires no unnecessary surgery.* This board just plugs in with no strap or additional connections. In spite of its quality, the Ramex-16™ costs just $139.95, complete with a *one year* limited warranty.

A complete turnkey memory management system on a disk — using either one or two 16K cards. HIDOS™ loads DOS onto one RAM card and with the second card loads an alternate language onto another. SOLIDOS™ turns a 16K card into a fast, 45-sector disk-drive emulator. At your local dealer or direct for just $34.95.



If you use VisiCalc™, then you must have THE CONSOLIDATOR. It will save you hours of keyboard time, by allowing you to manipulate totals of separate files *without reentering them.* Easy to use, invaluable to own. Just $49.95 at your dealer or direct.

All three of these together help make your Apple a more complete business system — giving you expanded memory, extra convenience, sure control. MasterCard and Visa holders order toll-free, 1-800-835-2246.

OMEGA MICROWARE, INC. T.M.

222 SO. RIVERSIDE PLAZA
CHICAGO, IL 60606
312-648-4844

Apple is a registered trademark of Apple Computer, Inc.
VisiCalc is a registered trademark of Personal Software, Inc.

CALL —A.P.P.L.E. TECHNIQUE:

IMBEDDING MACHINE CODE IN APPLESOFT PROGRAM FILES

William J. Cheeseman

THERE are at least four techniques commonly in use to "imbed" machine code routines in an Applesoft program file: (1) Linear POKEing, (2) READing a DATA statement and POKEing in a loop, (3) the S.H. Lam string method, and (4) the "transparent" method of inserting machine code in an artificial gap created between the end of the program and the end of program pointer. See *"All About Applesoft," Call –A.P.P.L.E. In Depth No. 1, pp.29–30.*

While in a sense the seed of the machine code routine resides in the Applesoft program, these four techniques have in common the feature that the final machine code resides at some location outside of the Applesoft program itself. The first three allow selection of any unused memory location, such as page 3. The last requires that the machine code, generated independently, appear immediately following the Applesoft program in memory. In none of them is the Applesoft program modified in any way, but instead continues to reside in memory in its original form. The machine code is not really "imbedded" in the Applesoft program.

Each of these techniques has some disadvantages. The first three waste memory, in the sense that the Applesoft routines which generate the machine code normally remain in memory, unused, after they have completed their task of loading the machine code. Furthermore, if you want to use a lot of small code routines, appropriate locations can soon be in short supply. The "transparent" method does not suffer from these shortcomings, but it is singularly uninformative to anyone examining the Applesoft listing — there is no hint that machine code lurks in the file. Also, if editing of the Applesoft program is contemplated, relocatable machine code must be used. The machine code must be generated by an assembler or some other device extrinsic to the Applesoft program.

This article discusses a modification of the DATA statement READ/POKE approach, which overcomes these disadvantages. It is based on the realization that a READ/POKE loop may be allowed to overwrite the DATA statement and related routines without interrupting execution of the generating loop. Thus, the DATA statement itself, which is used only once, *can become the final location of the machine code routine.* There will always be enough room, because each element of the DATA statement occupies at least two bytes (a digit and a comma), but generates only one byte of machine code. In this manner, the Applesoft program can be loaded and listed to reveal the machine code generating routines, but no space is wasted when the machine code they generate is POKEd into the midst of the Applesoft program file. This makes available a virtually unlimited number of memory locations for short, relocatable machine code routines, which do not interfere with use of page three and other desirable scratch areas. Furthermore, if the machine code is POKEd into the beginning of the Applesoft program at $801, nonrelocatable code can be used.

This technique is possible because the READing of DATA statements is controlled by a zero page pointer which references absolute memory locations, automatically incrementing as each DATA element is READ. Once the pointer has found the DATA token and moved beyond it, the DATA token and following elements can be overwritten without interrupting the functioning of the pointer. The pointer always stays ahead of the POKEd machine code.

All of the examples listed and discussed below create a trivial machine code routine for demonstration purposes: JSR $DB5A, JMP $D7D2. When CALLed by the program, it executes an Applesoft internal routine which prints a question mark to the screen, then calls a routine which continues execution of the next Applesoft statement.

1) At the Beginning

The first example, Listing #1, generates the machine code at the beginning of the Applesoft program, then raises the start of program pointer over it. This has the apparent effect of "deleting" the first few Applesoft lines, which will be modified as machine code is POKEd into them. No effort to preserve the integrity of line numbers and line number pointers in this pre-empted area is needed, because Applesoft line searches will always start with the raised start of program pointer. Subsequent program listings will not show the modified area, which will appear to have disappeared (much like the "transparent" method).

Raising the start of program pointer over an area at the beginning of the program is not new. See "Pointers on Pointers", *Call –A.P.P.L.E. vol. V, no. 1, p. 75.* However, I have introduced one refinement which not only buries the unneeded lines, but also buries the undertaker, so to speak. Line 30 of Listing #1 uses the OLDTXT pointer at $79, $7A (dec 121, 122) to locate the pointer to the next line, POKEs the latter pointer directly into the start of program pointer, TXTTAB, at $67, $68 (dec 103, 104). Thus, subsequent listings will show the program as beginning at the next line, line 40. Lines 1 through 30 will have disappeared from view. This is cleaner than the method described in "Pointers on Pointers", which left the "undertaker" line visible as the first line of the modified program. In addition, it avoids an apparent bug which may cause the "Pointers on Pointers" technique to fail on page boundaries.

IMBEDDING CODE IN APPLESOFT FILES

Placing the machine code at the beginning of the program has the advantage of allowing use of nonrelocatable code, since the code will always start at $800 (assuming the Applesoft program is loaded at the normal location). CALL 2048 will always execute the machine language routine. In addition, this is at the beginning of a page of memory. With careful counting of bytes, additional nonrelocatable routines can be stacked up in this area. The self-modified Applesoft program can be rerun without reloading as often as desired — the machine code need not be regenerated because it is already there from the first run. Note that the start of program pointer and the null byte at $800 should normally be reset before loading another program — either by POKEing from an exit routine in your program, POKEing manually, typing "FP," or rebooting. However, if you wish to use the same machine code routines with the next program, you can simply load it without resetting the pointer.

This same technique should work with the S.H. Lam string method as well, since the real string is usually in high string memory space by the time the POKE loop begins to overwrite the program line which creates the string variable assignment.

2) Scattered Hither and Yon

Alternatively, the machine code routines and their generating DATA statement READ/POKE loops can be scattered throughout your Applesoft program. This requires a little more overhead, but may be useful in complex programs. It has the advantage of leaving the start of program pointer in its normal location. However, under most circumstances, relocatable code must be used, and the location of the code must be calculated and preserved in a variable so it can be CALLed later. Because this variable would be cleared, the program cannot be rerun unless it is first reloaded. (The value of the variable, however, may be preserved by POKEing the high and low bytes into an unused zero page location and subsequently restoring the variable in a second program with PEEKs.)

The "before" version of Listing #2 demonstrates the technique. Line 30 is the key. (Lines 10 and 110 could just as easily have been incorporated directly into line 30, but they should be subroutines if you have more than one machine code routine to generate.) Line 30 uses FN ADR(121) to calculate the address of the null byte at the end of the prior program line, using the pointer OLDTXT at $79, $7A (dec 121, 122). An offset (in this example, 9) is added to place the beginning of the machine code routine, A, well into the line (for reasons to be described). The programmer must supply the correct offset, as well as the length, L, of the machine code routine, which is needed to control the READ/POKE loop. Note that variables A and L will have the correct values to be used with BSAVE and other DOS file commands. In addition, the programmer must supply the proper negative offset in line 110 (in this case, -4).

Because of the way Applesoft executes, it is necessary to provide a means for the program to bypass the machine code after it has been generated, before the program circles back to re-execute this section. (If this section will not be executed again, the remainder of this discussion can be ignored and the housekeeping routines can be omitted.) The problem is this: Once the machine code has been generated, on a subsequent pass through this section, the program would try to interpret it as Applesoft statements and generate a syntax or logic error.

Three solutions are possible. One is to POKE the Applesoft token for REM (178) into the first byte of the line. Then subsequent passes will ignore the rest of the line, thinking it is a comment. This method does work; it obviates the need to calculate offsets; and it is shorter. However, I dislike the approach. For one thing, a listing of the program after execution is unsightly, since the machine code bytes are printed as a "comment" after the REM. They appear as a combination of garbage and the unmodified balance of the Applesoft statements. Worse, the machine code is expanded in the listing because it is interpreted as if it were Applesoft tokens, and the resulting line might be longer than the maximum permissible Applesoft line.

A better approach (or at least more pleasing) is to POKE a GOTO statement aimed at the next line, so the rest of the line will simply be bypassed on subsequent passes. It is easy enough to POKE the GOTO token (171) and a null byte (0) or ASCII colon (58) signalling the end of the statement, but how to generate the right line number? This can be done with Applesoft routines, but the overhead is extensive. The gimmick used here is to start the line with the statement "IF [next line number] THEN . . ." Thus the GOTO token can be POKEd over the IF token, and the null byte or colon can be POKEd over the THEN token. The correct line number remains in place, as supplied originally by the programmer. Applesoft treats any positive integer as a Boolean "true" value, so the IF . . . THEN statement will cause the rest of the line to be executed the first time through. On the second pass, it will have become a GOTO . . . , which will bypass the (now modified) remainder of the line.

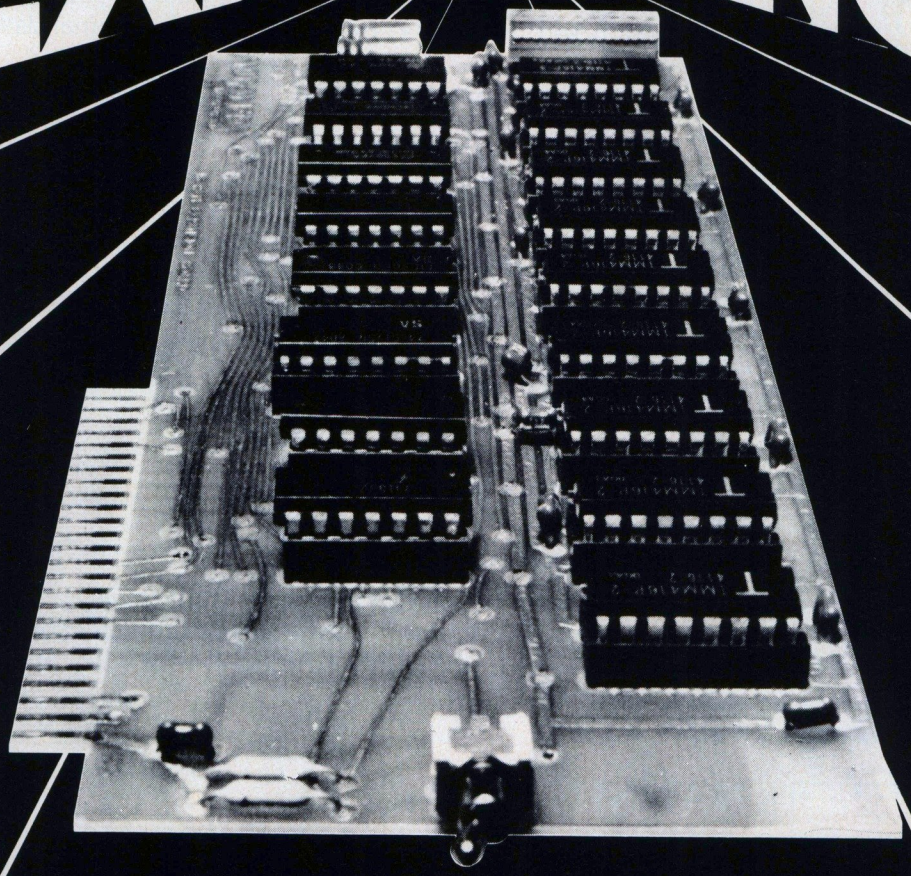
This leaves one task, to accommodate line numbers of differing lengths. The proper offsets must be supplied by the programmer, as described above. Bearing in mind that the defined FN ADR (121) returns the memory location of the last byte of the prior line, the locations of the needed POKEs can be counted on this basis: add four bytes for the program line number and the pointer to the next line, add one byte for the IF token, add one byte for each digit in the next line number, add one byte for the THEN token, and add one byte to get to "A", where the machine code starts. The negative offset in line 110 is obtained by counting backwards from "A" to the IF token, where a GOTO token will be POKEd.

POKEing a null byte in place of the THEN token fools the Applesoft listing routine into believing it is the end of the line. Thus, listing the program after it has been run hides the machine code. See the "after execution" version of Listing #2. If the colon (ASCII 58) were used, the technique would still work but the machine code would list as garbage, just as it does if REM is used.

You can see that the machine code is present by editing, adding or deleting a line in the program after it is run once, then listing it again.

NEW LOW PRICE
DIRECT FROM MANUFACTURER
$120.00

EXPAND ☆ ENHANCE



16K RAM EXPANSION BOARD FOR THE APPLE II* $120.00

The Andromeda 16K RAM Expansion Board allows your Apple to use RAM memory in place of the BASIC Language ROMs giving you up to 64K of programmable memory. Separate Applesoft* or Integer BASIC ROM cards are no longer needed. The 16K RAM Expansion Board works with the Microsoft Z-80 card, Visicalc, DOS 3-3, Pascal, Fortran, Pilot, and other software. A switch on the card selects either the RAM language or the mainboard ROMs when you reset your Apple.

The Andromeda 16K RAM Expansion Board has a proven record for reliability with thousands of satisfied customers.

Now with One Year Warranty.

ANDROMEDA
☆ INCORPORATED
Greensboro, NC. 27410
P.O. Box 19144

919 852-1482

VISA

master charge
THE INTERBANK CARD

Price for Andromeda 16K RAM expansion board now only $120.00. Please add $5 for shipping and handling. North Carolina residents add 4% sales tax.

*DEALER INQUIRIES WELCOME.

Time and Money

Our software saves you both.

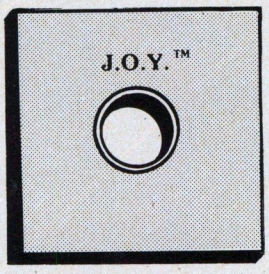
THE DEAN'S LIST™

List Study Aid

Makes memorizing simple and fun while it quizzes you on facts. Completely disk based. Includes complete set of disk commands and instruction book.

DOS 3.2, Apple][* or Apple][Plus*, 32k RAM, 1 drive

$24.95



J.O.Y.™

Simultaneous Printing Utility

Allows your Apple* to print and compute at the same time. Works with almost any printer, card, and slot. Prints either text files or Apple-Soft* program files. DOS 3.3 and 48k RAM

$39.95

APPLE-CILLIN II™

Hardware Diagnostic System

Checks for glitches in hardware quickly and easily. Verifies over two dozen hardware operations. New features include disk speed test, linked tests, printed results. Apple][* or Apple][Plus*, 16k minimum RAM

$49.95

NANSII™

The Information Manager

Manages virtually any kind of information. Advanced features and comprehensive capacity, but still easy to use. Special demonstration package $29.95.

CP/M°, 56k RAM

Specify 8" Standard CP/M° or 5¼" Apple CP/M° Disks

$249.95

Call or write us today for more specific information. You may order from your dealer, or call us toll free at 800-233-7512. In Pennsylvania call 717-243-5373. Or write us at XPS, Inc., 323 York Road, Carlisle, PA 17013. We'll be glad to send you our catalog of innovative software, free. (Pennsylvania residents add 6% sales tax.)

XPS, Inc.

Quality Microcomputer Software
323 York Road, Carlisle, PA 17013
(800) 233-7512 / (717) 243-5373

MasterCard VISA

*Trademark of Apple Computer, Inc.

°Trademark of Digital Research, Inc.

IMBEDDING CODE IN APPLESOFT FILES

This regenerates the linked list of line pointers, and Applesoft interprets the first few bytes of machine code as a line number. See line 19675 of the "after editing" version of Listing #2. For this reason, it is important to be careful while developing or editing any program which uses these tricks. It can be helpful to insert a REM at the beginning of the line during development, to prevent it from executing.

Finally, if the machine code generator is placed in a subroutine, the IF...THEN gimmick and all of these offset calculations can be avoided. Merely POKE a RETURN token into the first byte of the line and a null into the second byte, and POKE the machine code starting with the third byte. Then on subsequent passes the call to this subroutine will simply return without doing anything. See Listing #3.

Note that the S.H. Lam string technique cannot conveniently be used in this "hither and yon" fashion, since you normally will not know what starting memory location to specify. If you are willing to calculate the memory location once editing has been completed, however, it should work.

```
]LIST

1  REM

LISTING 1
10  DATA  32,90,219,76,210,215
20  FOR I = 2048 TO 2053: READ B:
      POKE I,B: NEXT
30 PTR =  PEEK (121) +  PEEK (122
    ) * 256 + 1: POKE 103, PEEK
    (PTR): POKE 104, PEEK (PTR +
    1)
40  REM

[THE REST OF YOUR PROGRAM]

50  CALL 2048
60  PRINT : PRINT "THE TEST WORKE
    D IF YOU SEE A '?'
70  END
```

```
]LIST

1  REM

LISTING #2 (BEFORE)

*** READ POINTER FUNCTION ***
10  DEF  FN ADR(X) =  PEEK (X) +
      PEEK (X + 1) * 256
20  REM

[THIS IS A CHUNK OF YOUR PROGRAM]

30  IF 40 THEN A =  FN ADR(121) +
    9:L = 6: DATA  32,90,219,76,
    210,215: GOSUB 110: REM  (TH
    IS LINE WILL BE MODIFIED)
40  PRINT "A = "A", L = "L: CALL
    A
50  PRINT : PRINT "THE TEST WAS S
    UCCESSFUL IF YOU SEE A '?'"
60  REM

[THIS IS MORE OF YOUR PROGRAM]

70  END : REM

*** CODE POKER SUBROUTINE ***
110  FOR I = A TO A + L - 1: READ
     B:POKEI,B: NEXT : POKE A - 4
     ,171: POKE A - 1,0: RETURN
```

```
]LIST
LISTING #2 (AFTER EXECUTION)
*** READ POINTER FUNCTION ***
10  DEF  FN ADR(X) =  PEEK (X) +
      PEEK (X + 1) * 256
20  REM

[THIS IS A CHUNK OF YOUR PROGRAM]

30  GOTO 40
40  PRINT "A = "A", L ="L: CALL A

50  PRINT : PRINT "THE TEST WAS S
    UCCESSFUL IF YOU SEE A '?'"
60  REM

[THIS IS MORE OF YOUR PROGRAM]
70  END : REM

*** CODEPOKER SUBROUTINE ***
110  FOR I = A TO A + L - 1: READ
     B: POKE I,B: NEXT : POKE A -
     4,171: POKE A - 1,0: RETURN
```

```
]LIST

1  REM

LISTING #2 (AFTER MORE EDITING)

*** READ POINTER FUNCTION ***
10  DEF  FN ADR(X) =  PEEK (X) +
      PEEK (X + 1) * 256
20  REM

[THIS IS A CHUNK OF YOUR PROGRAM]

30  GOTO 40

19675 SGN  SCRN( (121) + 9:L = 6:
      DATA  32,90,219,76,210,215:
      GOSUB 110: REM  (THIS LINE
     WILL BE MODIFIED)
40  PRINT "A = "A", L = "L: CALL
    A
50  PRINT : PRINT "THE TEST WAS S
    UCCESSFUL IF YOU SEE A '?'"
60  REM

[THIS IS MORE OF YOUR PROGRAM]
70  END : REM

*** CODEPOKER SUBROUTINE ***
110  FOR I = A TO A + L - 1: READ
     B: POKE I,B: NEXT : POKE A -
     4,171: POKE A - 1,0: RETURN
```

Get MORE out of VISICALC With

V-UTILITY $99.95

COPYRIGHT © 1981 YUCAIPA SOFTWARE — APPLE II DOS 3.2

V-UTILITY CONSISTS OF THE FOLLOWING APPLE II PROGRAMS THAT EXPAND THE USE OF VISICALC.

ALL PROGRAMS AVAILABLE SEPARATELY $39.95 EACH

V-OVERLAY — PROVIDES THE FOLLOWING SELECTION OF OVERLAYS TO LOAD ON TO THE VISICALC SHEET. MOVING AVERAGES, EXPOTENTIAL SMOOTHING EQUATIONS, STOCK AND COMMODITY TREND EQUATIONS, DATE COLUMNS FOR DAYS MARKET IS OPEN. A REAL TIME SAVER.

V-PLOT — THIS PROGRAM AUTOMATICALLY INPUTS DATA FROM A VISICALC COLUMN, PERFORMS AUTO SCALING THEN PLOTS EITHER 1 OR 2 COLUMNS ON A REGULAR LINE PRINTER (GRAPHICS PRINTER NOT REQUIRED). IDEAL FOR QUICK ANALYSIS OF UP TO 254 NUMERICAL DATA POINTS.

V-STAT — PROGRAM COLLECTS DATA AUTOMATICALLY FROM THE VISICALC COLUMN AND CALCULATES NUMERICAL DISTRIBUTION, CORRELATION COEFFICIENT, REGRESSION ANALYSIS, CHI 2 TEST AND T-TEST. YOU SELECT THE COLUMNS FOR DATA ENTRY AND SPECIFY THE ROW# TO START AND ROW# TO END DATA COLLECTION.A VALUABLE PROGRAM FOR MANUFACTURING, ENGINEERING, EDUCATION, AND STOCK ANALYSIS. PROGRAM IS EASY TO OPERATE.

V-PRINT — WITH THIS PROGRAM YOU MAY SELECT THOSE COLUMNS YOU WANT TO PRINT AND PLACE THEM IN ANY ORDER YOU SELECT. A VERY FLEXIBLE VISICALC PRINTING UTILITY.

YUCAIPA SOFTWARE

12343 12TH ST · YUCAIPA·CA·92399

PHONE (714) 797-6331

APPLE II is a Trademark of APPLE COMPUTER INC
VISICALC is a Trademark of PERSONAL SOFTWARE INC

DOS 3.2 OR DOS 3.3

IMBEDDING CODE

```
]LIST

1  REM

LISTING #3

*** READ POINTER FUNCTION ***
10  DEF  FN ADR(X) =  PEEK (X) +
     PEEK (X + 1) * 256
20  REM

[THIS IS A CHUNK OF YOUR PROGRAM]

30  GOSUB 90
40  PRINT "A = "A", L = "L: CALL
    A
50  PRINT : PRINT "THE TEST WAS S
    UCCESSFUL IF YOU SEE A '?'"
60  REM

[THIS IS MORE OF YOUR PROGRAM]
80  END : REM

*** 'UNDERTAKER' SUBROUTINE ***
90 A =  FN ADR(121) + 7:L = 6: DATA
      32,90,219,76,210,215: GOSUB
     110: RETURN : REM  (THIS LIN
     E WILL BE MODIFIED)
100  REM

*** CODEPOKER SUBROUTINE ***
110  FOR I = A TO A + L - 1: READ
     B: POKE I,B: NEXT : POKE A -
     2,177: POKE A - 1,0: RETURN
```


DOUBLETIME PRINTER™

IT'S LIKE GETTING ANOTHER APPLE FOR ONLY $150!

DOUBLETIME PRINTER (D.P.) is an extremely thorough and extensive package, which can easily pay for itself in a matter of weeks in computer time savings.

Until now, whenever the Apple was outputting information to a printer, it was "out of commission" until the printing was done. Because most printers are rather slow, this can mean a loss of use of the computer ranging 5 to 10 minutes to an hour or more.

D.P. now liberates your Apple from being "printer-bound" by allowing the computer to essentially do two things at once. With **D.P.** installed you'll be able to continue using many programs in the "foreground" while the printer faithfully prints out the desired files in the "background".

D.P. is more than just a simple interrupt driven utility though. Over a year of development has gone into producing a complete and integrated package with a wide variety of functions and features. A few of these are:

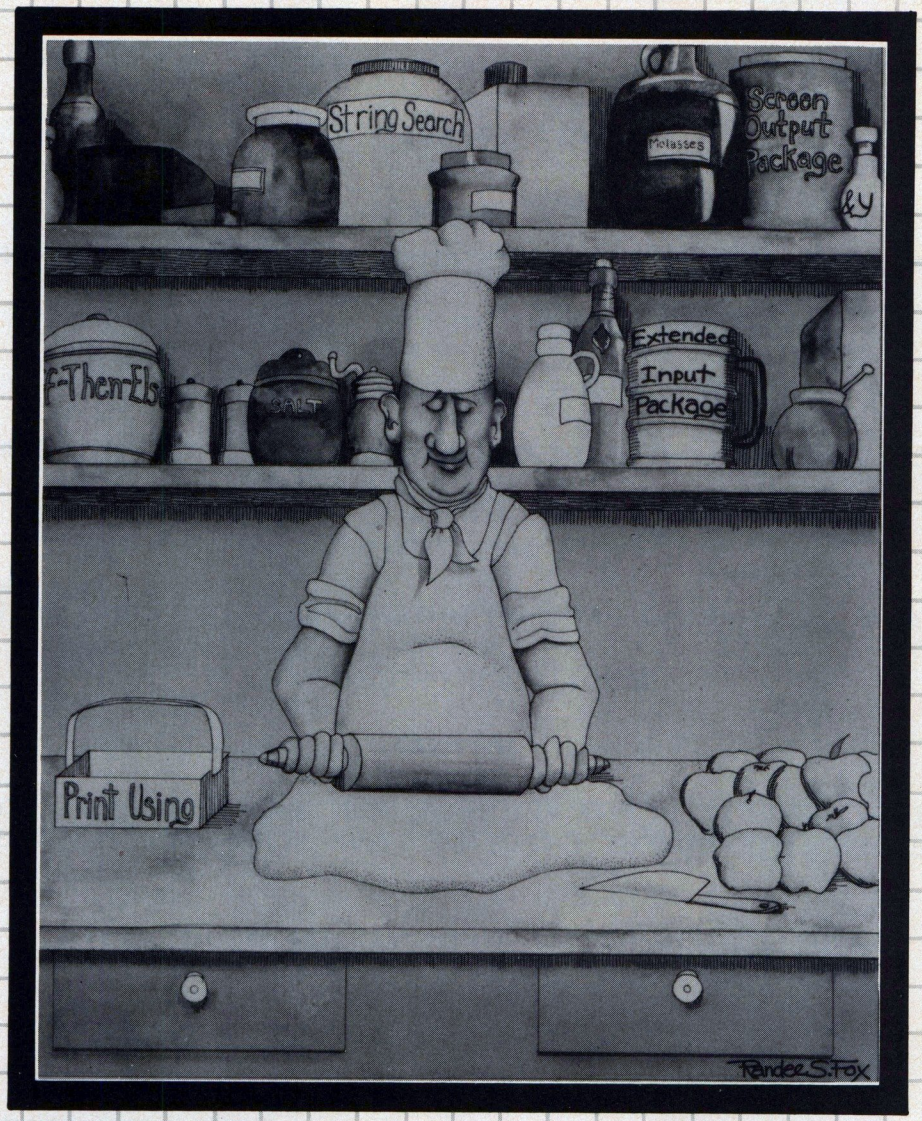
- Prints BINARY, TEXT or APPLESOFT files with no conversions necessary. All files are printed directly from the diskette so very little internal computer memory is used, and there are no restrictions on number or size of the documents printed, other than your system's disk capacity.
- Files can be FORMATTED when desired to include margins, paging and even variable number of copies.
- Files can be prioritized so that other users can add their files to a diskette while printing is in progress. Urgent files can supercede earlier files placed on the disk.
- **DOUBLETIME PRINTER** is supplied with a special F8 ROM (under special license from Apple Computer, Inc.) and an interrupt driver interface card. Both are simple to install by either end-user or dealer.

SYSTEM REQUIREMENTS: 48K Apple II/II+ with **DOS 3.3**

Apple, Apple II, Apple II+ are trademarks of Apple Computer, Inc.

southwestern data systems™

P.O. BOX 582 • SANTEE, CALIFORNIA 92071 • 714/562-3670



The extended Basic for your Apple.

APPLE SPICE

$29.95

042-0133
APPLE 2 DISK 48K

By: Corey Kosak & David Fox

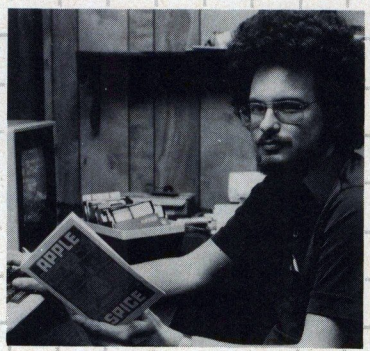
ALL YOU NEED TO EXTEND YOUR APPLESOFT BASIC.

APPLE SPICE is a set of fast, assembly language routines that alleviate some of the drudgery of writing programs in Applesoft BASIC. APPLE SPICE allows you to concentrate more on your program logic and less on things like "prettifying" the screen and detecting keyboard entry errors. APPLE SPICE also greatly expands Applesoft's capabilities by adding features found in many larger BASICs such as PRINT USING, STRING SEARCH (INSTR), IF-THEN-ELSE, and LINE INPUT.

- Easy-Patch - No machine language knowledge is necessary — beginners can use it.
- Excellent tutorial step by step manual. Written in easy to understand English.

APPLE SPICE is not copy protected. You may include APPLE SPICE in your commercial programs without additional charge. Just mention you used APPLE SPICE on your title screen and in your documention.

APPLE SPICE comes with a clearly written hands-on manual (52 pages) that explains every feature in detail. All of the examples listed in the manual are included on the APPLE SPICE diskette, as well as some user-friendly programs that make it a snap to change APPLE SPICE'S parameters and to relocate APPLE SPICE to where YOU want it in memory.



"Let your Apple II take care of the details so you can just program!"

Adventure INTERNATIONAL
A DIVISION OF SCOTT ADAMS, INC.
BOX 3435, LONGWOOD, FL 32750 • (305) 862-6917

AVAILABLE FROM YOUR FAVORITE DEALER OR CALL TOLL FREE (800) 327-7172
SHIPPING & HANDLING EXTRA, PRICES SUBJECT TO CHANGE

A9 — 0930

A WORD PROCESSOR FOR $5.00

Well, almost!

SANDY WORD PROCESSOR

ONE OF THE OLDEST WORD PROCESSING PROGRAMS FOR THE APPLE

HAS FINALLY COME TO THE THE U.S. AND

SOFTWARE & WORD PROCESSING OF TEXAS

HAS EXCLUSIVE WORLDWIDE DISTRIBUTION RIGHTS TO THIS SOFTWARE

$5.00

WILL BRING YOU A DEMO PROGRAM WHICH SHOWS THE EASE OF USING

SANDY WORD PROCESSOR

AND

AS AN INTRODUCTORY OFFER...AFTER YOU HAVE RUN THE DEMO YOU CAN BUY THIS PROGRAM FOR THE ABSOLUTE LOWEST PRICE OF ANY COMPARABLE WORD PROCESSING SOFTWARE.

SANDY IS COMPATIBLE WITH VISION 80 BY VISTA
SOON TO BE COMPATIBLE WITH VIDEX (FREE UPDATE)

YES, UPDATES ARE FREE!

YOU ONLY PAY FOR THE POSTAGE.
SEND YOUR CHECK FOR $5.00 AND WE WILL SEND YOUR DEMO DISK BY RETURN MAIL.

NO EXTRA HARDWARE EXCEPT THE APPLE, MONITOR, ONE DRIVE AND A PRINTER
NO Z-80 CARD.

WE RETAIL THE SOFTWARE THROUGH OUR WORD PROCESSING COMPANY FOR $389.00 AND THIS INCLUDES MANY UTILITY PROGRAMS SUCH AS "SORT MAIL LIST", "MAILER" AND "LABELLER".

CALL A.P.P.L.E. SPECIAL....$279.00

AUTHORIZED DEALERS:

YE OLDE SOFTWARE SHOPPE
P O BOX 1558
RICHARDSON, TX 75080

MICROAPPLICATION
1605 W. DIVISION
ARLINGTON, TX 76012

SOFTWARE & WORD PROCESSING
420-D S. CARROLL DENTON, TX 76201 (817) 566-3888

DEALER INQUIRIES INVITED

CALL —A.P.P.L.E. TECHNIQUE:

DYNAMIC LOADING AND LINKING

John L. Shanks

THERE are a number of ways to link machine code to Applesoft Basic programs, and some of these were discussed in the June, 1979 issue of *Call –A.P.P.L.E.* ("Hiding Out in Basic," by Val J. Golding, p. 5, and "The Basic–6502 Linker," by Andy Hertzfield, p. 7). In particular, I would like to present an extension of the technique of appending machine code to the top of the Applesoft program. This is also called the "transparent" method, and is further discussed by Val Golding in *Call –A.P.P.L.E. in Depth,* no. 1, pg. 30, as well as William Cheeseman in this issue.

By putting the machine code on top of the BASIC, and then changing the various "top–of–program" and "beginning–of–variables" pointers, the machine code becomes a part of the BASIC. Hence, the machine code is saved and loaded with the BASIC. Also, the BASIC editor will automatically move the machine code up and down in memory when changes are made to the BASIC statements. This points out one problem with this technique. The machine language code must be relocatable, since you have no control over where it is loaded, and it may be moved after it is loaded. However, there are coding techniques to overcome this restriction, and I believe the procedure is quite useful.

The extensions in this article allow you to load and link any number of machine code files to an Applesoft program. Each file may have any number of entry points, or other locations which must be accessed from BASIC. Since the linking is dynamic and occurs at the time the BASIC program is run, the entry points are picked up automatically. Also, the machine code may be maintained separately from the BASIC code, and no changes are required to be made in the BASIC code when updated machine code files are loaded. Furthermore, the machine code files may be either executable code or data files, such as "shape tables."

The code used in this procedure consists of a short *loader* program, written in BASIC, which may be embedded in the BASIC code, a *linker* routine which is run as a part of the initialization of the BASIC program, and a machine code *preamble* which is appended to each machine code file to be loaded.

This procedure hinges on the one-line BASIC subroutine,

```
63999 F1 = PEEK(121) + 256
      *PEEK(122) + 34: RETURN;
```

which is a slight modification of the code presented by Mark Cross in his article "Appending Machine Language to BASIC," p.60 of the June, 1981 issue of *Call –A.P.P.L.E.* This must be the last statement of the program, as it is used to find the first available free memory location after the last BASIC statement. F1 is the location at which we wish to load our first machine code file.

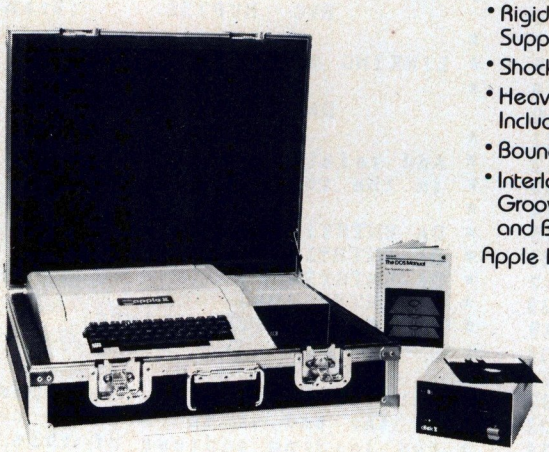
Preamble

It is desirable to be able to load a number of machine code files, with the first being loaded at the address F1 found by subroutine 63999. The second one, of course, should be loaded right above the first, and the third right above that, and so on, so we need to know the length of each file loaded. Also, this information has to be available to the running program so it knows where to link into the machine code. For this purpose, I propose to use a machine code routine at the beginning of each file, the "preamble" mentioned above. The purpose of the preamble is twofold; first, to move the end-of-program and beginning-of-variable pointers from the beginning of the machine code file to the end of the machine code file, and secondly, to provide locations where the BASIC program can find the information it needs to access the machine code.

FLIGHT FORM

BY CASES, INC.

DESIGNED TO PROTECT YOUR COMPUTER



Features —

- Rigid Shell Made of Plywood Supported High Impact ABS.
- Shock Resistant Foam Lining.
- Heavy Duty Hardware Includes Key Locking Latches
- Bound Metal Edges.
- Interlocking Tongue and Groove Extrusion, Mating Lid and Bottom.

Apple II Plus 2 Drives $175.00

OTHER MODELS AVAILABLE

CLUB AND DEALER DISCOUNTS AVAILABLE

CASES, INC. P.O. Box 33820 Seattle, WA 98133 (206) 365-5210

DYNAMIC LOADING AND LINKING

Listing A shows the preamble. The MVPTRS routine fetches the end-of-program address from the EOP location, adds the length of the machine code, and stores the result in the locations EOP, SVSTRT, ARSTRT, AND ENSTOR. These locations may be found in several references, including *Call -A.P.P.L.E. in Depth,* referenced above, and the publication *What's Where in the Apple*, by W. F. Luebbert. Each machine code file should contain this code. The operational code should be inserted between the MVPTRS routine and the FILEND = * statement. With the Big Mac Assembler, it is a simple matter to append the operational source code to the preamble, and move the FILEND = * statement to the end of the program.

One other change is required in the assembler program. Following the DA = FILEND —ORG should be a list of entry points for that file in the form

DA ENTRY1 — ORG
DA ENTRY2 — ORG
DA ENTRY3 — ORG, etc.

These are the addresses of all the entry points in this file, relative to the origin, or load point of the file. There may be up to sixty-two of these entry point addresses in the table, but the first item in the table must be DA = FILEND — ORG.

LOADER

Given that every file to be loaded now contains the preamble code, we can now design the loader. An example of a loader is shown in lines 63990-63993 of Listing B. In this example, four machine code files are being loaded. Since we want the first file to be loaded at the address right above the last BASIC statement, the first statement executes a "GOSUB 63999," which sets F1 equal to that location. Next, since we know that the preamble requires that the "top-of-program" pointer contain that same address, it is POKEd into locations 175 and 176. Then the first file is "BLOADed" from the disk to that location. Instead of the "file 1 name", use the actual name of the machine code as it is saved on disk.

Now that the file is loaded, it is necessary to move the program pointers. This is done by the machine code preamble which is CALLed at location F1. However, the variable F1 has been destroyed by BLOADing the machine code file, so it is necessary to pick up that address by CALLing the value previously saved at locations 175 and 176.

If there were only one file to load, this would be the only statement required in the loader. Statement 63991 shows how the next file is loaded. Since the CALL to the preamble moves the pointers to the top of the first file, it is sufficient to simply BLOAD the second file to the address now contained in locations 175 and 176. A CALL to that address executes the preamble in the second file and moves all the required BASIC pointers. This continues for all the machine code files which need to be loaded.

```
ASM
                        1   *
                        2   *      *** LISTING A ***
                        3   *
                        4   * PREAMBLE FOR LOADING AND LINKING
                        5   * MACHINE CODE TO BASIC
                        6   *
                        7   *
                        8   SVSTRT   =    $69
                        9   ARSTRT   =    $6B
                       10   ENSTOR   =    $6D
                       11   EOP      =    $AF
                       12   *
                       13   ORG      =    *
                       14   *
8000: B8               15   START    CLV              ;Reset the V bit
8001: 50 02            16            BVC  MVPTRS      ;Branch always
                       17   *
                       18   * LINKING ADDRESSSES TABLE
                       19   *
8003: 1F 00            20            DA   PRGEND-ORG
                       21   *
                       22   * Add relative addresses of entry points here
                       23   * in the following format
                       24   *
                       25   * DA ENTRY1-ORG
                       26   * DA ENTRY2-ORG
                       27   * DA ENTRY3-ORG
                       28   * DA ENTRY4-ORG
                       29   * (UP TO 62 ENTRY POINTS)
                       30   *
                       31   *
                       32   * This routine moves the Applesoft program pointers
                       33   * to the end of the machine code. It assumes that
                       34   * the loading address of this file is available in the
                       35   * End-Of-Program (EOP) location
                       36   *
                       37   *
8005: A5 AF            38   MVPTRS   LDA  EOP         ;Get the machine code starting LOC
8007: 18               39            CLC              ;as loaded by BASIC
```

```
8008:  69  1F     40                ADC  #<PRGEND-ORG ;add the code length
800A:  85  AF     41                STA  EOP         ;put in Applesoft EOP location,
800C:  85  69     42                STA  SVSTRT      ;simple variable start address,
800E:  85  6B     43                STA  ARSTRT      ;array space start address, and
8010:  85  6D     44                STA  ENSTOR      ;end of numeric storage address.
                  45  *
8012:  A5  B0     46                LDA  EOP+1       ;Get the Hi byte
8014:  69  00     47                ADC  #>PRGEND-ORG ;add the Hi byte
8016:  85  B0     48                STA  EOP+1       ;and save it
8018:  85  6A     49                STA  SVSTRT+1
801A:  85  6C     50                STA  ARSTRT+1
801C:  85  6E     51                STA  ENSTOR+1
801E:  60         52                RTS              ;Return to calling program
                  53  *
                  54  * The operational user code goes here
                  55  *
                  56  *ENTRY1 XXX XXXX
                  57  *        ...
                  58  *        ...
                  59  *        RTS
                  60  *
                  61  *ENTRY2 XXX XXXX
                  62  *        ...
                  63  *        ...
                  64  *        RTS
                  65  *
                  66  *ENTRY3 XXX XXXX
                  67  *        ...
                  68  *        ...
                  69  *        RTS
                  70  *
                  71  *ENTRY4 XXX XXXX
                  72  *        ...
                  73  *        ...
                  74  *        RTS
                  75  *
                  76  *
                  77  PRGEND   =    *
```


BUSINESS AUTHORS WANTED

(APPLE AND IBM P.C.)

If you have written Business Software for the Apple Computer, or the IBM personal computer, Westware, Inc., would like to hear from you.

Westware, Inc., offers national advertising, experienced documentation writing, and software support. If you are interested in nationally marketing your business product, write:

WESTWARE SOFTWARE, INC.
2455 S. W. 4th Avenue, Suite #2
Ontario, OR 97914

Supercharge Your APPLE II*

The Axlon RAMDISK™ 320K Memory System for the Apple II and Apple II Plus* provides access speeds never before available. The Axlon memory system is designed to interact with Apple DOS 3.3* and Apple Pascal 1.1* like two standard floppy disk drives while delivering the lightning fast access speeds of RAM memory. This also leaves 32K of RAM for advanced programming techniques. The interface board is slot independent and draws no power from your Apple. The rechargeable battery system built into the unit provides three hours of backup in the event of a power loss. Drop by your local Apple dealer or contact Axlon, Inc. for more information.

* Trademark of Apple Computer, Inc.
* Pascal is a Trademark of U.C.S.D. Regents

- Plug-in compatibility
- 320K bytes of RAM (200NS) memory designed to function like two 35 track floppy disk drives
- Compatible with Apple DOS 3.3 and Apple Pascal 1.1
- Same size as the Apple Disk II* Drive
- Invisible memory refresh - even with the Apple turned off
- Rechargeable battery system built-in to provide 3 hours of auxiliary power
- Slot independent interface board - draws no power from your Apple
- All firmware is in static RAM on the interface board
- Includes software for diagnostic, fast load and copy routines, and business applications

AXLON

170 N. Wolfe Road,
Sunnyvale, CA 94086
(408) 730-0216

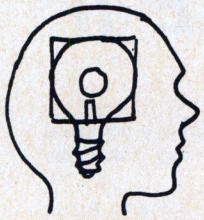
DYNAMIC LOADING AND LINKING

It is important to note that the loader is terminated by an "END" statement. The loader is a stand-alone program. It is only run to initially load the machine code files on top of the BASIC code. Thereafter, the BASIC may be saved, loaded, changed, etc., and the machine code files will still be a part of the program. The loader need be run again only if there has been some change in the machine code, or if for some reason the machine code section of the BASIC program gets clobbered. If no such occurrences are anticipated, the loader code may be deleted from the program after the machine code files have been loaded.

LINKER

The linker code is a part of the BASIC program and must be run somewhere in the program before the machine code is required by the BASIC. Listing B also shows an example of a linker.

ATTENTION
SOFTWARE AUTHORS



CDS

Cortland Data Systems

Let our team of professionals design, package and distribute your software creations. Royalties paid monthly.

If you have a quality software product, call or write us today for a free authors kit or send us a machine readable copy of your work. All submitted programs will be reviewed in strictest confidence.

P.O. Box 14414
Chicago, IL. 60614
Telephone: 312-929-7727

The statement at 10000 just defines a double word PEEK function to cut down on having to type so many asterisks and parentheses. At 10010, the GOSUB 63999 finds the beginning of the first machine file, which was loaded there by the loader. To find the beginning of the second file, we need to know the length of the first file. Recall that this information is the first item of the address table of the preamble of machine File 1. We know where that is. It must be three locations up from the beginning of File 1. Therefore, by PEEKing at the double word at F1+3, we get the length of File 1. Hence, the address of the second file can be obtained from

```
F2 = F1 + FN PK(F1+3)
```

Thus we continue to find the beginning of each of the machine code files which were loaded by the loader.

After finding the beginning of each file, we need to know the location of the various entry points in each of the files. As an example, it was assumed that File 1 had two entry points, File 2 had four entry points, and both Files 3 and 4 each had only one entry point. Lines 10050-10120 show how these addresses are obtained.

We will assume that the entry points are listed in the linking address table in ascending order, as shown in Listing A. Each address requires two bytes, so the relative address of Entry1 of File 1 will be found at F1+5, Entry2 at F1+7, etc. Lines 10050–10060 pick up the two entry points in the machine File 1.

As we indicated above, File 2 has four entry points, and line 10070-10100 get the addresses of those. These are all entry points in machine code File 2, so the base address is F2 now instead of F1. Lines 10110 and 10120 get the single entry points for files 3 and 4.

Listing B illustrates a particular example, but in general, if FS(J) indicates the starting address of machine File J, and EP(I,J) indicates the Ith entry point of File J, then the following equations hold:

```
FS(1) = F1 (From GOSUB 63999)
FS(J+1) = FS(J) + FN PK(FS(J)
+ 3), FOR J = 1,2,3,...,
EP(I,J) = FS(J) + FN PK(FS(J) +
3 + 2*I).
```

Finally, there are extensions to this procedure which could be considered. For example, the loader could be made into an EXEC, rather than put right into the program. Also, if sufficient memory space is saved above the BASIC program, dynamic loading could be used by loading in only those machine code files which are needed, at the time they are called by the BASIC code.

SUMMARY

The procedure presented here provides a general approach to loading and linking machine code to Applesoft. It tends to modularize the program development, so that the BASIC programmer does not need to become directly involved in the machine code development. He only needs to know the names of the machine code files, and the order in which the entry points are listed in the address linking tables. On the other hand, the machine coder can develop and maintain the assembler programs without too much concern about impact on the BASIC program (unless functional specificatons change). The new machine code, after being loaded by the special loader, is automatically linked when the BASIC program is run. The complete program can be saved and used until new machine code needs to be loaded.

To implement the system, the BASIC programmer needs to add the line 63999 to the end of his program, and write the loader and linker for his application. The machine coder needs to append the preamble code to each of his files, making sure the FILEND = * statement is at the end of each, and that all the entry points are entered into the address linking tables.

The following steps will put the loader and linker into operation:

1. Load the BASIC program.
2. Insert the disk containing all the machine code files to be loaded.
3. Run the loader program in the BASIC code.

Now the program can be run, saved, loaded, and/or changed, without any further need to run the loader.

LISTING B

```
10  REM

LOAD AND LINKER SUBROUTINES
Appending machine files to FP

      by  John L. Shanks
Call -A.P.P.L.E. : July 1982

9999  REM

LINKER SUBROUTINE

To be run as part of the
program initialization

10000  DEF  FN (PK)X =  PEEK (X) +
     256 *  PEEK (X + 1): REM "

Find the starting addresses for
each machine file

10010  GOSUB 63999
10020 F2 = F1 +  FN PK(F1 + 3)
10030 F3 = F2 +  FN PK(F2 + 3)
10040 F4 = F3 +  FN PK(F3 + 3): REM

Find all the entry points
file 1... two entry points

10050 E1 = F1 +  FN PK(F1 + 5)
10060 E2 = F1 +  FN PK(F1 + 7): REM

File 2... four entry points

10070 E3 = F2 +  FN PK(F2 + 5)
10080 E4 = F2 +  FN PK(F2 + 7)
10090 E5 = F2 +  FN PK(F2 + 9)
10100 E6 = F2 +  FN PK(F2 + 11): REM

Files 3,4... one entry point each


10110 E7 = F3 +  FN PK(F3 + 5)
10120 E8 = F4 +  FN PK(F4 + 5): RETURN

63989  REM

LOADER SUBROUTINE

63990  GOSUB 63999: POKE 176,F1 /
     256: POKE 175,F1 - 256 *  PEEK
     (176): PRINT  CHR$ (4)"BLOAD
      FILE 1 NAME,A";F1: CALL  PEEK
     (175) + 256 *  PEEK (176)
63991  PRINT  CHR$ (4)"BLOAD FILE
      2 NAME,A"; PEEK (175) + 256
      *  PEEK (176): CALL  PEEK (
     175) + 256 *  PEEK (176)
63992  PRINT  CHR$ (4)"BLOAD FILE
      3 NAME,A"; PEEK (175) + 256
      *  PEEK (176): CALL  PEEK (
     175) + 256 *  PEEK (176)
63993  PRINT  CHR$ (4)"BLOAD FILE
      4 NAME,A"; PEEK (175) + 256
      *  PEEK (176): CALL  PEEK (
     175) + 256 *  PEEK (176): END

63999 F1 =  PEEK (121) + 256 *  PEEK
     (122) + 34: RETURN
```


LOGO
© M.I.T.

Krell's LOGO for APPLE II*
Includes: our new low price and

1. Two copies of Krell's LOGO for Apple II*
2. A UTILITY disk containing M.I.T.'s extremely valuable demonstration material, a series of pre-defined shapes which may be substituted for the LOGO TURTLE, and Krell's timesaving library of pre-defined procedures.
3. ALICE IN LOGOLAND, a twenty program tutorial series for intermediate and experienced programmers new to the LOGO language.
4. The technical manual LOGO for Apple II* by H. Abelson & L. Klotz. This is the official technical manual issued by M.I.T.
5. LOGO FOR THE APPLE II* by Harold Abelson, Byte Books. A complete instructional manual for intermediate and advanced users of LOGO.
6. THE ALICE IN LOGOLAND PRIMER, a step by step, lavishly illustrated introduction to LOGO for those who are new to both programming and to LOGO, (available in May and sent free to all purchasers of Krell's LOGO for Apple II*).
7. A comprehensive wall chart that portrays, explains and graphically illustrates the LOGO commands in action.
8. A one-year's free subscription to the LOGO & EDUCATIONAL COMPUTING NEWSLETTER. (Regularly $30.00 per year). A new authoritative source of information about the structure, classroom application and capabilities of LOGO, plus an up to the minute forum on the most significant new ideas and issues in educational computing.

$149.95

College Board SAT* Prep Series

TRS-80, APPLE, PET, ATARI, CP/M, PDP-11 Based on past exams, presents material on the same level of difficulty and in the same form used in the S.A.T.

S.A.T., P.S.A.T., N.M.S.Q.T. — Educator Edition set includes 25 programs covering Vocabulary, Word Relationships, Reading comprehension, Sentence Completion, and Mathematics. $229.95
INDEPENDENT TESTS OF S.A.T. SERIES PERFORMANCE SHOW A MEAN TOTAL INCREASE OF 70 POINTS IN STUDENTS' SCORES.

GRADUATE RECORD EXAM Series — Educator Edition includes 28 programs covering Vocabulary, Word Relationships, Reading Comprehension, Sentence Completion, Mathematics, Analytical Reasoning and Logical Diagrams. $289.95

Micro-Deutsch

MICRO-DEUTSCH set includes 24 grammar lessons, covering all materials of an introductory German course. Four test units also included. Grammar lessons use substitution transformation drills, item ordering, translations and verb drills. Drill vocabulary based on frequency lists. Suitable for use with any high school or college textbook. Extensively field tested at SUNY, Stony Brook. APPLE*, PET*. $179.95

Isaac Newton + F.G. Newton

Perhaps the most fascinating and valuable educational game ever devised - ISAAC NEWTON challenges the players to assemble evidence and discern the underlying "Laws of Nature" that have produced this evidence. Players propose experiments to determine if new data conform to the "Laws of Nature". FULL GRAPHICS NEWTON - presents all data in graphic form. This game is suitable for children. Players may select difficulty levels challenging to the most skilled adults. Both Games $49.95

Krell Software Corp.

"The State of the Art in Educational Computing"

1320 STONY BROOK ROAD / STONY BROOK NY 11790 / (516) 751-5139

Krell Software Corp. has no official ties with the College Entrance Examination Board or the Educational Testing Service. Krell is, however, a supplier of products to the E.T.S. N.Y.S. residents add sales tax.
PROGRAMS AVAILABLE FOR THE TRS80, APPLE II, PET & ATARI.
*Trademarks of Apple Comp Corp., Tandy Corp., Commodore Corp., Atari Corp.

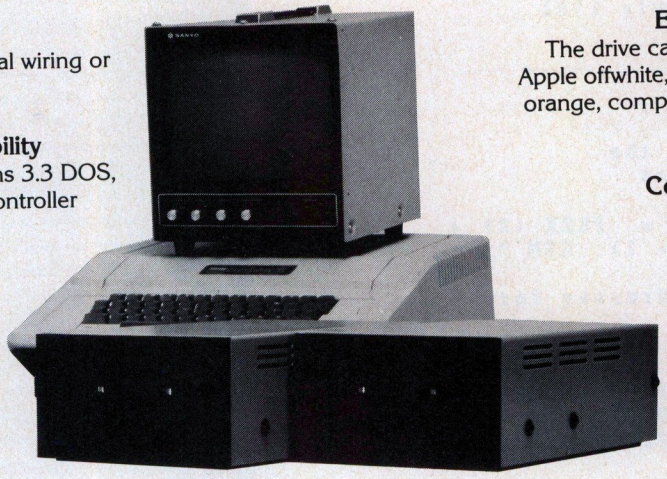
Introducing low cost, Apple II compatible disk drives

40-track drive with half-tracking for only $375.00

Easy to install
Simple plug-in with no additional wiring or power supply required.

Complete Apple II compatibility
40-track, 5¼ inch drive that runs 3.3 DOS, PASCAL or CP/M (Apple disk controller required).

Full Warranty and Service
90-day warranty plus service center for out-of-warranty service.



Eight colors to choose from
The drive cabinet is available in a standard Apple offwhite, lime green, dark green, bright orange, computer blue, brilliant yellow, black or chrome.

Complete Disk Drive System
For only $375, you get the 5¼ inch disk drive, color coordinated cabinet, and cable. Or, there's a dual drive system that includes dual 40-track disk drives, cabinets, Apple disk controller, and cables for only $850.00.

For further information, or to order the Apple II compatible disk drives, call or write:

I² INTERFACE, INC.
7630 Alabama Ave., Unit 3
Canoga Park, CA 91304
(213) 341-7914

Dealer and quantity discounts available upon request
MasterCard, VISA or COD orders accepted. Apple and Apple II are registered trademarks of Apple Computer, Inc.

KORSMEYER ELECTRONIC DESIGN

TIRED OF COMPLICATIONS?

MENTOR, for your Apple][,][+ or /// computer, saves you time at the design bench by calculating a vast number of AC, DC, filter, communication, transistor & vacuum tube formulas – each in an instant. Quotients to these formulas are accompanied by HIRES graphics and tables whenever applicable. The user-friendly MENTOR program is completely menu-driven with single-letter commands which require no carriage return. Optional use of single-letter suffixes allows entry of values as milli, micro, meg, kilo & pico. The MENTOR manual includes listings of mathematical formulas, background information, required user input and more. *$124.95 + $6.25 shipping & handling.

SLICK SWITCH is the 40-to-80 column-mode toggle for Sup'R'Term owners. Change modes with just the flip of a switch – great for a quick change between Pascal text and graphic modes too. Installation is easy, requires no soldering and uses no I/O sockets. *$24.95 + $1.50 shipping.

QUICK SLOPE makes it possible for the alphaSyntauri musician to change between linear and exponential envelope modes in approximately one second. With this speed, you can use both modes in the same tune without the delay you've had to endure before now. QUICK SLOPE appends to your alphaPlus software quickly and easily, eliminating the need for any disc swapping. *$34.95 + $1.50 shipping.

Ask your local Apple Dealer to show you the MENTOR Demo Disc and for help in ordering any of our software/hardware line. *Direct sale terms are prepaid or C.O.D. – CA customers include 6% sales tax.

MENTOR, SLICK SWITCH & QUICK SLOPE – trademarks of Korsmeyer Electronic Design
Apple – a trademark of Apple Computer, Inc.
Sup'R'Term – a trademark of M & R Enterprises
alphaSyntauri & alphaPlus – trademarks of Syntauri Corp.

KORSMEYER ELECTRONIC DESIGN • 16411 Del Mar • Huntington Beach, CA 92649

DUAL DRIVE MASTER CREATE

David L. Deever

A nuisance problem with the DOS 3.3 MASTER CREATE program is that it insists on having the slave diskette in the same drive as the master. When I spend the bucks for a second drive I would like my software to use it effectively. The information in Worth and Lechner's *Beneath Apple DOS* on how to split MASTER CREATE into a loading segment and a writing segment (pp. 7-1,7-2) together with some searching to find the IOB table in MASTER CREATE (at $E4F) give rise to the following Integer Basic program for a DUAL DRIVE MASTER CREATE. The master diskette containing the MASTER CREATE program and the DUAL DRIVE MASTER CREATE program is the SOURCE DISKETTE. The slave diskette to be mastered is the TARGET DISKETTE.

Doing the same thing in Applesoft Basic is a bit tricky. The corresponding program would be the same except that line 10 is omitted and the commas in INPUT statemens become semicolons. However, memory from $800 up to nearly $4000 is required by MASTER CREATE. Since an Applesoft program normally begins at $801 it must be moved out of the way. There are several options. You may use the &LOMEM: utility *(Apple Orchard* Vol. 1, No. 1 or from ***Higher Text)*** or you may do a

```
POKE 103,1: POKE 104,64:
POKE 16384,0
```

before loading your program, or you may place the above statement in a small "boot" program followed by

```
PRINT CHR$(4)"RUN DUAL DRIVE
MASTER"
```

If you wish to change slot as well as drive, you should be able to POKE the slot-number * 16 (e.g., 96 for slot 6, 80 for slot 5, etc.) into 3664, but I haven't tried this.

```
]LIST

10  FOR I = 1 TO 127
20  POKE 50,I
30  PRINT "?";
40  NEXT
50  REM

 SO WHAT DID YOU EXPECT DEPT.

CHRIS VOLPE : CALL -A.P.P.L.E.
```

```
]LIST
    5 REM

  DUAL DRIVE MASTER CREATE

     by David L Deever

Call -A.P.P.L.E. : July 1982

   10 POKE 74,0: POKE 75,64: POKE
      204,0: POKE 205,64: REM
LOMEM: 16384

   20 TEXT : CALL -936
   30 PRINT "DUAL DRIVE MASTER CREATE"
      : PRINT
   40 INPUT "SOURCE DRIVE: ",S: IF
      S<1 OR S>2 THEN 40
   50 INPUT "TARGET DRIVE: ",T: IF
      T<1 OR T>2 THEN 50
   60 PRINT "BLOAD MASTER CREATE"
      : REM
BEGINS WITH CTRL-D

   70 POKE 2061,76: REM
SPLIT INTO TWO PARTS

   80 POKE 3665,S: REM
SET SOURCE DRIVE

   90 CALL 2048: REM
EXECUTE DOS LOAD

  100 POKE 3665,T: REM
SET TARGET DRIVE

  110 CALL 2093: REM
EXECUTE DOS WRITE

  120 END
```


SUPER RAM II
16K Language Card
With ROM SOCKET
2 Yr. Warranty
$125.

COMPUTER PRODUCTS
DESIGNING
MANUFACTURING
ELECTRONIC ENGINEERING

R. H. Electronics Inc.
COPYRIGHT © 1981 - PATENTS PENDING
566 Irelan, Bin CA
Buellton, CA 93427
(805) 688-2047

SUPER FAN II ™
FOR YOUR APPLE II COMPUTER*
$74.95

APPLE OWNERS

"COOL IT"

- CLIPS ON—NO HOLES OR SCREWS • REPLACEABLE SWITCH
- AVAILABLE IN 120V OR 240V AND 50/60HZ • DURABLE MOTOR
- REDUCES HEAT CAUSED BY EXTRA PLUG-IN CARDS
- INCREASED RELIABILITY—SAVES DOWN TIME AND REPAIR CHARGES
- TAN OR BLACK COLOR
- LOW NOISE DUE TO DRAWING EFFECT OF AIR THROUGH YOUR COMPUTER AND SPECIAL FAN AND MOTOR DESIGN
- "QUIETEST FAN ON THE MARKET"
- EXTRA 120V OUTLET FOR MONITOR OR ACCESSORIES TURNS ON WHEN YOU TURN ON YOUR FAN (NOT AVAILABLE ON 240V MODEL)
- SOLD WORLD WIDE • UNIQUE 1 YEAR WARRANTY

COPYRIGHT © 1981 - PATENTS PENDING

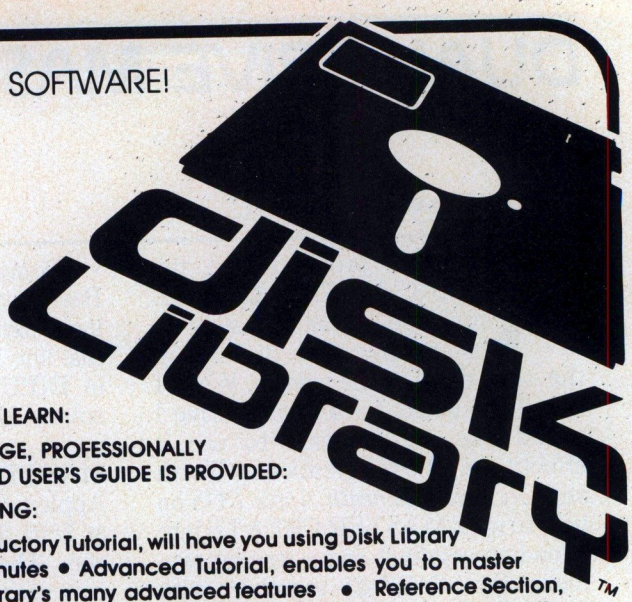
HOW TO HOOK IT UP: 1) Clip it on your APPLE. 2) Unplug your 120V cable (you won't need it). 3) Plug short 120V cable from Super Fan II to the back of your computer. 4) Plug the supply cable from Super Fan II to your 120V power source. 5) Turn on the rocker switch and a built-in, red, ready light comes on. 6) You are all set to "COOL IT." This switch also turns your computer "off" and "on."

Dealer Inquires Invited

*Plus 6% California Sales Tax and $2.50 shipping charge per item.

TAKE CHARGE OF YOUR COLLECTION OF DISK-BASED SOFTWARE!

THE SOFTWARE MANAGEMENT SYSTEM



DISK LIBRARY is an elegant, user-oriented system for creating and maintaining a thorough, cross-referenced index of all your disk-based programs and data files. It provides for AUTOMATIC entry into your library file of the full catalog of any Apple* diskette. Disks formatted under other operating systems (such as Pascal and CP/M*) are easily entered from the keyboard. Written entirely in machine code, DISK LIBRARY'S operation is both smooth and swift.

EASY TO OPERATE:
• Menu-driven • User-definable prompt defaults • Single keystroke operation • Full featured Editing • Super fast Sorts by any field (1200 items sorted in 4 seconds!) • Works with all disks created under DOS 3.1, 3.2 and 3.3 • User definable Program Types (e.g., Business, Game, Utility) of up to 15 characters each can be assigned to each program entry with single keystrokes or via block actions • On-screen and printed Summaries, by File type (Integer, Applesoft, Binary, Text) and by Program Type (e.g., Accounting, Graphics, Music) • Block Actions (global editing/deleting) • Instant Searches . . . by full or partial string (find any item in 1/3 sec.!) • New Files can be Appended to existing records, in memory or on disk • Unique Feature: User can redefine the Disk Volume Number displayed by the DOS Catalog Command • A Unique Volume Identifier and Disk Title can be Assigned to each disk entry in your library file • Printed Reports are attractively formatted for easy readability

EASY TO LEARN:
A 75 PAGE, PROFESSIONALLY PREPARED USER'S GUIDE IS PROVIDED:

INCLUDING:
• Introductory Tutorial, will have you using Disk Library in 10 minutes • Advanced Tutorial, enables you to master Disk Library's many advanced features • Reference Section, provides quick answers for experienced users • Applications Section, gives you many ideas for maintaining your library • Index, enables you to find whatever you need

SYSTEM REQUIREMENTS: 48K Apple II or II+ with DOS 3.3

DISK LIBRARY is licensed by MODULAR MEDIA

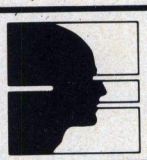
Suggested Retail Price $59.95

*Apple, Apple II and Apple II+ are registered trademarks of Apple Computer, Inc. CP/M is a registered trademark of Digital Research, Inc.

southwestern data systems™

P.O. BOX 582-C • SANTEE, CALIFORNIA 92071 • 714/562-3670

ANSWER YOUR TELECOMMUNICATIONS NEEDS WITH HELLO CENTRAL!



HELLO CENTRAL! from Advanced Operating Systems changes your Apple II® or Apple II Plus® into a highly-versatile communications center. With any acoustic or direct-connect telephone modem and HELLO CENTRAL!, you can put your Apple in direct communication with other distant computer systems—mainframes, minis, and other micros. You can also easily tap into the wealth of information available from any of the hundreds of public data-base services like THE SOURCE.

Establish your own electronic mail system. Create an automatic phone-dialer directory. Produce, edit, manipulate, and print text files. Instruct your Apple to answer the phone, send out a greeting, and save any incoming messages to disk. You can do all of these things and more with HELLO CENTRAL!

Text can be uploaded and downloaded using text buffering. The buffer provides a temporary storage space, holding up to 18,000 characters. Complete buffer control allows you to easily alter any portion of the buffered file before transmission or SAVEing.

The best thing about HELLO CENTRAL! is that you don't have to be a professional to get professional results.

HELLO CENTRAL! is designed to work with any Apple II® or Apple II Plus® with Applesoft® in ROM and 48K. Only one disk drive is required.

Hook up with HELLO CENTRAL! today, and say HELLO to a whole new realm of computing power!

Now available at your local software retailer, or call (800) 348-8558 to order. (Indiana residents, call (219) 879-4693). MasterCard and VISA accepted. $99.

Advanced Operating Systems
450 St. John Road
Michigan City, IN 46360

*Apple II, Apple II Plus and Applesoft are registered trademarks of Apple Computer, Inc.

APPLEMASH:

IN AMID$T STRINGS

Michael Thyng

ARE you still interested in MID$ and IF? To start with did you know that you don't even need MID$? That's right because we can use LEFT$ and RIGHT$. See example 4.

```
]LIST

7  REM EXAMPLE 4
10 X$ = "THE FASTIDIOUS WIND UNCO
     RKED THE MAGNUM OF BUBBLY FR
     OM THE NAPA VALLEY."
20 M1$ =  LEFT$ (X$,63)
30 M2$ =  RIGHT$ (M1$,30)
40 M3$ =  RIGHT$ ( LEFT$ (X$,63),
     30)
50 M4$ =  MID$ (X$,34,30)
200  PRINT
210  PRINT M1$
215  PRINT
220  PRINT M2$
225  PRINT
230  PRINT M3$
235  PRINT
240  PRINT M4$
```

If you had an alphanumeric variable X$ as we have in line 10, and you wanted to extract the characters "MAGNUM OF BUBBLY FROM THE NAPA," you could just use a combination of LEFT$ and RIGHT$.

There are at least two ways to do it. One way is to follow the pattern I used in line 20. After line 20 is executed, M1$ = "THE FASTIDIOUS WIND UNCORKED THE MAGNUM OF BUBBLY FROM THE NAPA." In line 30 we complete the maneuver with RIGHT$. We could have obtained the phrase we wanted in one line as shown in line 40. More conventionally we would use line 50. They all work. Sometimes in a program we are trying to locate a specific character. In example 5, we are trying to locate the middle initial for people who have written their names on a petition. Some of the people wrote their names last name first. Others wrote them first name first. Our job is to locate the initials.

What distinguishes an initial from the first letter of a name? Well, I think in this case, all initials will be followed immediately by a period. So, if I test for a period, find one and then take the character to the immediate left of the period, I should have an initial.

```
]LIST

7  REM EXAMPLE 5
10  INPUT "ENTER NAME ";NA$
20  FOR K = 1 TO  LEN (NA$)
30  IF  MID$ (NA$,K,1) = "." THEN
     100
40  NEXT K
50  PRINT "THERE WAS NO INITIAL"
60  END
100 I$ =  MID$ (NA$,(K - 1),1)
110  PRINT I$
```

In this example, the data is all input using an input statement, but certainly names could have been read in from a disk file. Line 20 sets up the parameter of the test. We want to look at every character from 1 to the full length of NA$. So, no matter how long NA$ is, the loop starting in line 20 will cover all the characters. The loop starting in line 20 will allow us to check out each of the character.

Line 30 is the promised code. Here we want to know if the current character equals a period. IF is the required word to initiate a test. MID$ is the string operation that allows us to look at the middle portion of an alphanumeric variable.

Here we are looking at one character at a time of the variable NA$. As K varies from 1 to the length of NA$, we get to test each character. If (or should I say when) we detect the period, we branch to line 100. In line 40, the switch is flipped, so to speak, for K to increase in value by 1 until K equals the number of characters in NA$. If it should happen that all the characters are tested and there is no period, then we reach the end of the FOR...NEXT loop.

When that happens, we drop through to lines 50 and 60 where we stop. Under normal conditions, where an initial and a period exist, we will get to line 100. Once there, we will capture the initial by moving one character to the left of the period (K–1).

While this routine is short and only tests for one character, it shows how to use Applesoft to help you get the most our of your programming efforts.

A harvest of savings from

Apple Tree Electronics

SOFTWARE

APPLE · ATARI · TRS80 · IBM

A full line of software for business, games and education **up to 35% off!**

MUSE	IUS
VISICORP	STONEWARE
ON LINE	SYNERGISTIC
EDU-WARE	HAYDEN
HOWARD	AND MANY MORE

HARDWARE

AMDEK · HAYES · MICROSOFT

	List	Our Price
32K RAM card	$293.00	$205.00
Video Term	$345.00	$279.00
Lazer Products	– – – 20% off – – –	

DISKS

Maxell	Box of 10, 5¼", SS-DD	$35.00
Verbatim	Box of 10, 5¼", SS-DD	$29.00

MONITORS

LE MONITORS	List	Our Price
9" Green	$189.00	$159.00
12" Green	$199.00	$169.00
ZENITH		
12" Green	$179.00	$129.00

Plus a full line of **AMDEK** Monitors

PRINTERS

PAPER TIGER	List	Our Price
460G	$1,094.00	$950.00
560G	$1,394.00	$1,250.00
EPSON		
MX 70	$449.00	$395.00
MX 80FT	$745.00	$595.00
MX 100FT	$945.00	$795.00

CALL FOR THIS MONTHS SPECIAL!

1-800-835-2246 EXT. 211

OR

702-452-5589

VISA

5130 East Charleston Blvd.
Suite 5CA
Las Vegas, Nevada 89122

MasterCard

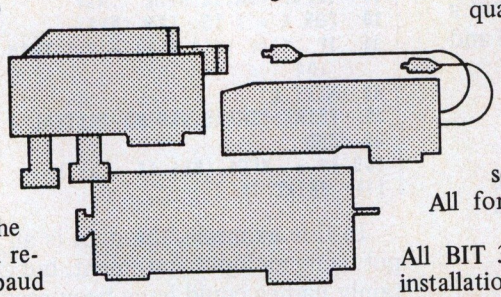
Phone orders welcome. Mail orders may send charge card number (include expiration date), cashiers check, money order or personal check (allow ten business days for personal or company checks to clear). Add $3.00 for shipping, handling and insurance. Nevada residents add 5.75% sales tax. Please include phone number. All equipment is in factory cartons with manufacturers warranty. Equipment subject to price change and availability. Call or write for price list.

apple add-ons

APPLE ADD-ONS are products by BIT 3 Computer Corporation to increase the capability of your APPLE II or APPLE II PLUS. Make it an 80-column word processor with the FULL-VIEW 80 . . . or a full-fledged terminal with the DUAL-COMM PLUS . . . or increase its memory with the MEMORY PLUS.

DUAL-COMM PLUS

adds two fully independent serial I/O interfaces to your APPLE II. Terminal mode firmware converts an APPLE into a full or half duplex computer terminal. With a modem such as the NOVATION CAT™ or the HAYES SMART-MODEM™ connected to one serial port and a printer connected to the other port you can use the APPLE as a remote terminal with online printer. 16 baud rates are available from 50 to 19200 baud. Price: $239.00



The FULL-VIEW 80 provides an 80 x 24 display yet retains the standard APPLE display too. You can switch between 80 and 40 column mode via keyboard or program. An 8 x 10 cell size character set provides quality characters, full descenders, and line-drawing graphics. The

FULL-VIEW 80

is compatible with APPLE peripheral cards, Basic, Pascal and several word processing systems. All for $295.00 – no extras needed.

All BIT 3 products feature easy plug-in installation and are designed with you, the user in mind. Contact your local APPLE dealer or our offices for more information.

The MEMORY PLUS

adds 16K of RAM memory for your 48K APPLE II or APPLE II PLUS. The addition of the MEMORY PLUS represents a 33% capacity increase in RAM memory and 40% more available RAM for disk users since APPLE DOS occupies almost 11K of memory. It is compatible with Integer Basic, Applesoft Basic, Pascal, Z80 Softcard, Fortran and other products such as Lisa 2.0 and Visicalc 3.3. Price: $149.00

BIT 3
COMPUTER CORPORATION

8120 PENN AVENUE SOUTH, SUITE 548
MINNEAPOLIS, MN 55431
(612) 881-6955

MasterCard VISA

Apple II is a registered trademark of Apple Computer, Inc.; Z80 Softcard is a registered trademark of Microsoft Consumer Products. D.C. Hayes Smart-Modem is a trademark of D.C. Hayes, Associates, Inc.; Novation CAT is a registered trademark of Novation, Inc.

KEEP YOUR APPLE* cool as a cucumber

How? . . . with a Cool-Mark II fan unit.

- *Protects against overheating caused by plug-in boards and prolonged use.*
- *Saves down time and repair due to overheating.*
- *Mounts easily **INSIDE** your Apple II* in seconds, without hardware.*
- *High performance airflow and low noise level.*
- *Precision ball bearings = long service life.*
- *One year warranty.*

Cool-Mark II Fan Unit

$34.95
Quantity discounts to user groups

See Your Authorized Apple Dealer or Order Direct

How To Order: Send money order or check for $34.95, plus $2 for UPS surface shipping anywhere in continental USA.
California residents add 6% sales tax.

Send money order for PROMPT delivery. Personal checks must clear before shipment.

Call Collect To:(213) 287-9945, COD orders accepted.

*Apple is a registered trademark of Apple Computer Inc.

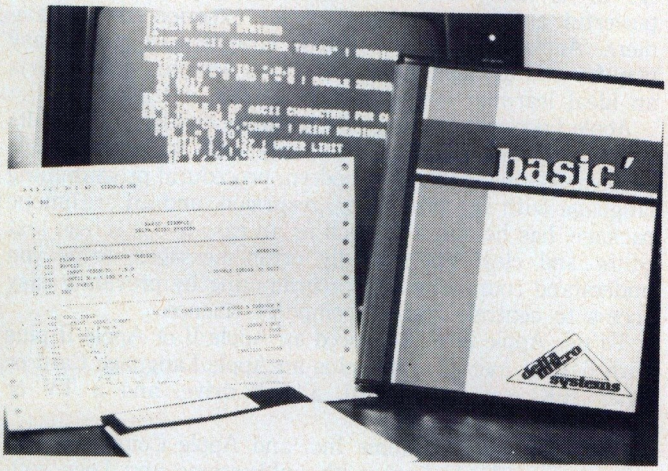
Dealer inquiries invited.

Mark Four Imports

P.O. Box A • San Gabriel, Ca. 91776 • (213) 287-9945

TOMORROW'S APPLES TODAY

BASIC'



Delta Micro Systems announces the release of Basic', a supremely logical programming environment. Basic' was developed in 1980 by Dr. J.L. Schmit as an "in house" tool and was subsequently used for all new system development by Computer Assisted Analysis—the developers of Compu-Trac and Intra-Day Analyst. At the urging of many colleagues it is now being made available to the public.

Basic' — a powerful, completely structured extension of the BASIC language — is an invaluable aid to the serious programmer. Blocks of code are indicated by a unique indentation convention, eliminating the need for both extraneous statements (such as BEGIN & END) and statement numbers. Comments are right justified for ease of reading, block indentation reveals the flow of control, procedures are separated by white spaces and an extensive cross-reference is provided.

Basic' provides full documentation and a program diskette in a handy binder format. It is now available for the APPLE, IBM PERSONAL COMPUTER and TRS/80 (other versions are under development) at $129 (plus postage, handling and applicable sales tax) from Delta Micro Systems, P.O. Box 15952, New Orleans, LA 70175, (800) 535-1814.

GL-PLUS — THE MANAGEMENT TOOL FOR BETTER ACCOUNTING

Great Divide Software, Inc. has introduced a general ledger accounting system for use on the Apple /// Computer called GL-Plus. It is a flexible, easy to use, journal-based general ledger system with a PLUS. The PLUS is a built-in accounts receivable and accounts payable capacity at no extra cost!

Features include: 1) Simple operation; 2) User specified chart of accounts; 3) Instant update of all accounts as each transaction is entered; 4) Large variety of reports; 5) Reports always as current as the last entry; 5) Built-in Account Receivable and Accounts Payable; 6) Can be used with a two disk drive system or with a hard drive drive.

For most managers accounting and the preparation of financial reports has been a time consuming and sometimes costly task. But now accounting procedures can be simplified! GL-PLUS, "The Management Tool for Better Accounting" is an accounting system designed for the Apple /// computer. It is a flexible, easy to use, journal based GENERAL LEDGER system. The computer and GL-PLUS combine to provide the user with a tool which will make accounting easier. GL-PLUS guides one quickly through journal entries and then posts them. Report preparation is a "snap" because GL-PLUS does it automatically. No more long hours of "balancing the books."

The GL-PLUS system is flexible to use. It will accept a chart of general ledger accounts using a numbering system of the users choosing (between 1 and 9999). The user can use existing account names and numbers or create a new one to fit their needs. In addition, GL-PLUS can easily support accounting for multiple companies. Each company's records are kept on separate data disks or different volumes on a hard drive and each can have a different chart of accounts. A menu option allows the user to quickly switch companies.

Both accounts payable and accounts receivable are easily implemented. The user decides if and when to use either or both. Each supports a subsidiary set of company records, invoice by invoice, for all current or outstanding balances. Receipts and payments can be applied to specific invoices, including partial receipts or payments.

The GL-PLUS accounting system is written for the Apple /// computer. It requires an Apple /// and a second disk drive or can be used wiht a hard disk drive. A printer with at least a 132 column capability is recommended, however, slightly abbreviated reports can be displayed on the screen, printed on an 80 column printer or on the Apple Silentype printer.

Available from Apple Computer Dealers or Great Divide Software, Inc., 8060 W. Woodard Dr., Lakewood, CO 80227, (303) 337-0383. Suggested retail price is $495.00.

VISISCHEDULE™ PROGRAM FOR APPLE III

VisiCorp™ has made available a more powerful version of its VisiSchedule™ software program for use on the Apple® III personal computer.

The new VisiSchedule program is an interactive project management program that takes full advantage of the advanced capabilities of the Apple III. These include the 80-character, upper- and lower-case display, expanded keyboard, greater internal memory and hard-disk data storage.

Using the VisiSchedule program, a person can create an on-screen schedule of projects and tasks to increase organizational efficiency. The displayed schedule responds immediately to "what if?" changes and shows the effects of the changes on all subsequent events. The VisiSchedule program updates information on-screen when a change is initiated and incorporates all pertinent data includes costs, available staffing, resource leveling and scheduling constraints.

TOMORROW'S APPLES TODAY

Because of the larger internal memory of the Apple III, the number of tasks that can be performed is nearly double that of the version available for Apple II. A 128K Apple III computer will support from 145 to 215 tasks per project, while the 256K Apple III will support 300 tasks per project. VisiSchedule projects can be stored on the Apple III's floppy or the Profile or Corvus hard-disk system and recalled when needed.

The timechart and other reports produced by the VisiSchedule program provide valuable information to any business involved in planning projects and managing resources. Managers, executives, engineers and manufacturing personnel can profitably use VisiSchedule to track any project in which deadlines and a schedule of events are important.

The VisiSchedule program requires an Apple III computer with a minimum of 128K memory and one external disk drive (either a floppy disk drive, Corvus hard disk drive or Apple ProFile hard disk drive). A printer is strongly recommended.

The suggested U.S. price of the VisiSchedule program is $300. It will be available in July. VISICORP, 2895 Zanker Road, San Jose, CA 95134, (408) 946-9000.

CRITICAL PATH SCHEDULING THE MANAGEMENT TOOL FOR BETTER PLANNING

Great Divide Software, Inc., has announced the CRITICAL PATH SCHEDULING SYSTEM, an aid in scheduling complex projects such as those in construction, aerospace, and many other industries. It is a powerful tool for analyzing and scheduling all the tasks required to complete any multi-level project ON TIME AND AT THE LOWEST COST.

Features include: 1) Simple operation; 2) Large project capacity (over 2000 tasks); 3) Easy update of project status; 4) Large variety of reports; 5) Graphical project presentation; 6) Provision for manpower planning; 7) Can be used with a one or two disk drive system or with a hard disk drive.

The use of the CRITICAL PATH SCHEDULING SYSTEM permits proper planning to insure that ALL important tasks are included at the start of the project and provides an overview of the entire project down to the lowest level task. It does this by using the information that the operator supplies for each task and then analyzing the whole project, from start to finish. It then compiles a report that lists every task, with start and completion times and determines which tasks are MOST critical (in the scheduling sense) to the completion of the project. For each task the AUTOMATIC DATE feature generates an early start and finish date, and a late start and finish date. These dates represent the earliest a task can be expected to start and end, and the latest the task can be postponed without affecting the completion time of the project. The difference between these two dates (called the "float") is shown on the report. Also shown is "Free Float" which is the amount of time a task can be delayed without affecting any other task or the completion time of the project. Armed with this kind of information the manager is better prepared to make decisions regarding the impact any task will have on any other task or the whole project and permits him to be instrumental in guiding the project rather than just monitoring its progress.

Available from Apple Computer Dealers or Great Divide Software, Inc., 8060 W. Woodard Dr., Lakewood, CO 80227, (303) 337-0383. Suggested retail price is $495.00.

LOGO COMPUTER SYSTEMS ANNOUNCES APPLE LOGO, AVAILABLE FOR THE APPLE II

Apple Logo is an exciting interactive computer language designed to be easy to learn and use. It encourages a beginner to do many simple and interesting projects, yet is a powerful and modern language for experienced programmers. Apple Logo feature Turtle Graphics—a quick, easy way for you to create pictures on your screen. Apple Logo is an ideal learning language for children and adults.

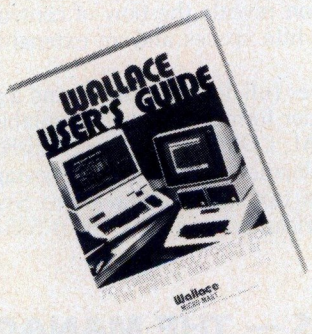
Apple Logo is a language for learning: children and adults write programs quickly and easily and acquire valuable problem-solving skills; is a modern, procedural programming language with list processing, recursion, and an arithmetic package; has helpful aids such as an easy-to-use screen text editor and clear and friendly error messages; helps you understand computers and programming by providing easy access to central ideas of computer science.

To use Apple Logo, you need an Apple II or Apple II plus with 64K of memory (48K plus an Apple Language Card or other 16K expansion card), a disk drive, and a TV or monitor.

Logo Computer Systems Inc. and Apple Computer, Inc. will support Apple Logo with publications and Logo software. Contact your Apple dealer and watch for more of our special mailings from Logo Computer Systems, Inc., 368 Congress St., Boston, MA 02210.

WALLACE MICRO-MART ANNOUNCES NEW CATALOG

Ron Wallace, President of Wallace Micro-Mart, Peoria, Illinois, recently announced the publication of his company's new catalog for Apple Computer software and accessories.



The "Wallace User's Guide" includes the latest programs and accessory items for Apple Computer system operators. The catalog, according to Wallace, currently has 1,500 items and is updated monthly with an average of 100 new items.

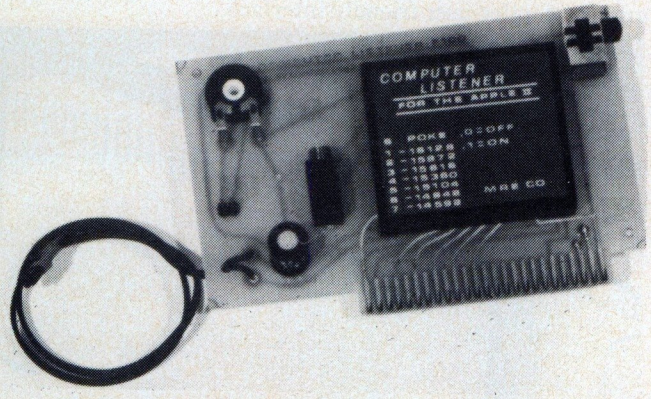
According to Wallace, the "Wallace User's Guide" can be used by corporations, dealers, clubs or enthusiasts. Plus, there are group discounts for quantity purchases. "There has been a great demand for our catalog since we began offering it," says Wallace. "This field is moving so quickly that we have to constantly update to include the state-of-the-art materials."

The catalog is available from Wallace Micro-Mart, 2619 N. University, for $5, (refundable with order).

HARDWARE:

THE COMPUTER LISTENER

The Computer Listener allows you to add a whole new dimension of sound to your programs. Now you can actually hear your Apple's internal activity just like in the movies. The Computer Listener adds sound effects to machine code games, math calculations, graphics, print statements, plotting, pokeing, listing, etc. Hear the disk save and load. All activity within the computer can be generated into unique sounds. The Computer Listener converts all the buss activity down to an audible level creating a variety of unique sounds. The internal activity is converted into a unique pulses which are then amplified and played through the Apple's Built-in speaker.



The Computer Listener can be used as a diagnostic tool to debug software and hardware problems; as a teaching device enabling the student to "see" the internal activity; or as a novelty device to give incredible arcade like sound to your programs.

SPECIFICATIONS: The Computer Listener plugs into any slot, can be activated in the software, the on board volume control adjusts loudness, plugs into Apple's built-in speaker as well as an external speaker jack, compatible with other plug-in boards.

Price $79.00. M. Reinhart Engineering Co., 4730 W. Addison, Chicago, IL 60641 (312) 286-6606.

INTELLIGENT DISK ASSEMBLY

If you're looking to add mass storage capacity to your Apple II without looking for the rich uncle to fund the enterprise, look no further. $999 is all you'll pay for our Intelligent Disk Assembly, which includes a state-of-the-art Winchester disk drive and an advanced Xebec single board controller. Add a few hundred dollars more and you'll also get the components to complete the subsystem. A highly reliable 115V/230V power module. A cable set. A host adapter personality card. Apple DOS, CP/M® or Pascal software and a component cabinet. Installation instructions and documentation make assembly simple. More important, it makes you a richer person because you do it yourself and save up to half the cost of comparable storage products.

Even if you have CPU attachments other than the Apple II, Xebec would like to hear from you. They are currently developing a whole range of small computer interfaces. For more information contact Xebec, 432 Lakeside Drive/ Sunnyvale, CA 94086. In California, call 800-672-1842, all others call 800-538-1644.

PROGRAMMABLE SERIAL I/O BOARD

A board fully compatible with current Apple® software including Pascal 1.1, Applesoft®, and Integer Basic. It provides a full EIA RS-22 DTE interface to most modems, printers and CRT terminals as well as an opto-isolated current loop interface strappable for full or half duplex, active or passive, 2-wire or 4-wire, 20mA or 60mA operation up to 200V.

On board 1K ROM software enables user programmable printer width control, delay after carriage return, automatic line-feed generation, and video display. Serial character format is user alterable with the most popular configurations set as default conditions.

Unique to the PSIO are a full range of continuously variable, programmable asynchronous data rates to 19.2 K baud; automatic recognition of the incoming data rate from a remote terminal; and firmware routines to pulse-dial calls onto the international Telex-TWX network.

Software programmable baud rates to 19200 BPS; Auto-baud: Automatic incoming baud rate recognition & matching; Telex/TWX pulse-dial function; 5-Bit baudot code, 7-Bit ASCII or 8-Bit EBCDIC capability; software break/send detect; CPU interrupt hardware/firmware. Price: $199.95. Intra Computer, 120-10 Audley Street, Kew Gardens, N.Y. 11415, (212) 947-5533.

SLOT 8 FOR APPLE II COMPUTER

— Legend Industries, Ltd. announces a space saver for Apple Computer users: Slot 8 for the Apple motherboard.

Are you finding that your Apple is overcrowded? If you need more space for your peripheral cards, consider Legend Industries as a haven for an inexpensive way to solve your problems. Legend has just introduced a product for your Apple II called "Slot 8." This product is just what the name reveals; slot number 8 for your Apple Computer.

Slot 8 is a card that plugs into slot seven and provides two (2) slots for peripheral use. Instead of slots 0-7, you now have slots 0-8!

Most of the peripheral cards for the Apple will work in slot 8, and this device is really a space saver. Price: about $50.00. Legend Industries, Ltd., 2220 Scott Lake Road, Pontiac, MI 48054.

OMNICOLOR 640 COLOR PRINTER

The Omnicolor 640 ink-jet printer provides professional quality color graphics at the lowest possible cost. Besides sharp, clear, high-resolution color graphics, the Omnicolor 640 is so quiet it can operate in any office environment.

Apple][...Apple ///...The IBM Personal Computer. Interface any of these popular computers to the Omnicolor printing system for only $4,995.

Crisp color contrast. Ink-jet technology provides excellent print quality in both the graphic and alphanumeric mode. When the Omnicolor 640 prints a solid color, it's solid.

Simplicity of operation. From screen to color copy in just a few keystrokes.

Features include: professional print quality, 64 vivid colors, low cost per copy, fast precision graphics, full-color alphanumerics, affordable, and available now from Quadram Corp., 4357 Park Drive, Norcoss, GA 30093, (404) 923-6666.

TOMORROW'S APPLES TODAY

WIZARD BUFFERED PRINTER CARD (FOR APPLE II)

The WIZARD Buffered Parallel Output (BPO) is a revolutionary new card designed to take the frustration out of normally time consuming printing jobs. With the WIZARD-BPO you will never again have to wait for your printer to finish a job before your computer is free to start on a new project. The WIZARD-BPO also unleashes all of your printer's built in capabilities... Block Graphics, Hi-Res Graphics Screen-Dumps, Full control of text formatting and page indexing. A personality ROM installed on your WIZARD-BPO tailors the operation to your particular printer. Personality ROM currently available for Epson printers, other personality ROMs available soon. Buffer sizes of 16K or 32 K available now.

The WIZARD-BPO is remarkable both for its advanced design and its ultra-reliable construction. Component leads, except for the RAMs and the firmware ROM, are all soldered directly to the board for the most reliable component interconnections. Low power consumption by the WIZARD-BPO also keeps things cool while presenting a minimum load on the power supply.

The WIZARD-BPO comes complete with a five-foot long cable for your printer, plus an owners manual that tells you how to enjoy all the BPO benefits. You also get a full two-year factory-replacement warranty from Wesper Micro Systems. 800-854-8737. Retail $179.00.

THE ONE CARD SOLUTION TO EXECUTE CP/M² APPLICATION PROGRAMS FOR APPLE II & III

APPLI-CARD™, 4 or 6 mhz Z-80 and 64K on-card memory.

The features include: CP/M available, SB-80 with card, 40 col. to 255 col. horizontal scroll, choice of application, 2K PROM on the card, real time clock available, expansion interface, upper and lower case capability, a self-contained Z-80A or Z-80B with memory, one-card Wordstar execution, 63K available for program development or execution, and menu driven set up. The prices for each product is: Wordstar APPLI-CARD, with 64K, 4mhz, Softvideo tm features SB-80 (CP/M compatible operating system), $545; Wordstar APPLI-CARD with 64K, 4mhz, Softvideo tm features CP/M is $595; APPLI-CARD with 64K, 4mhz, Softvideo tm features SB-80 (CP/M compatible operating system) is $395; APPLI-CARD with 64K, 4mhz, Softvideo tm features CP/M is $445; and Z80 Development APPLI-CARD with 64K, 4mhz, Softvideo tm features SB-80 (CP/M compatible operating system) includes PCPI-BASIC, Z80 assembler, linker is $495. These products are available from Personal Computer Products, Inc., 16776 Bernardo Center Drive, San Diego, CA 92128, (714) 485-8411.

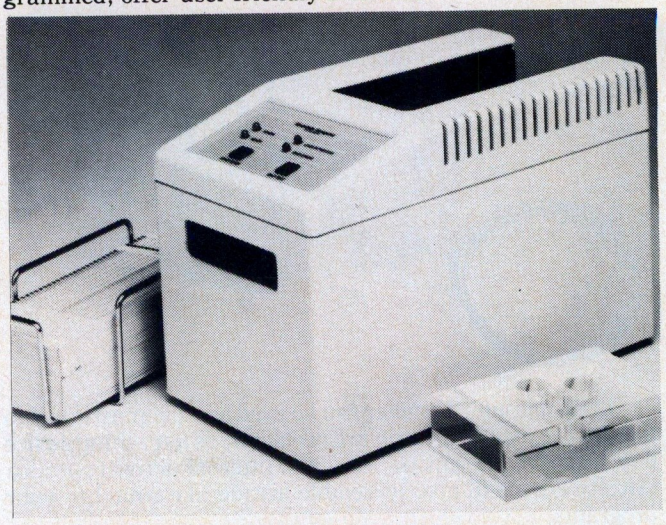
ACOLYTE

With the ACOLYTE you can have hardware control of APPLE memory and program flow. The features include: read and write to memory, trace program execution, interrupt and alter program flow, slow execution to a crawl, and halt on any program access to a specific address. All of this without stopping the program and without the program's 'knowledge' or co-operation. The price is $349.00 complete.

For more information write: Access Memory, 1701 West End Dr., Philadelphia, PA 19151.

CARDREADER FOR PERSONAL COMPUTERS

Mountain Computer Incorporated announces a low-cost, optical card reader for the personal and desk-top computer market. At $1,495.00, the MCI Model 1100A Intelligent Card Reader offers automatic card feeding with sophisticated features previously available only on much higher priced equipment. For example, the MCI Model 1100A can: read pencil marked and punched cards up to 14 inches in length, automatically feed 200 cards at a rate of 2 cards per second, distinguish marks from erasures, perform diagnostic self-test, send data remotely via a modem, provide multiple data conversion methods, including ASCII, Hexadecimal, Binary, numeric, and multiple choice grading, be remotely programmed, offer user-friendly interface.



The cost-effective MCI Model 1100A Card reader opens an exciting new world of applications for automatic data entry in the education, office and manufacturing environments. For additional information, contact: Mountain Computer, Inc., 300 El Pueblo Road, Scotts Valley, CA 95066 (408) 438-6650.

LPS II — HIGH RESOLUTION, HIGH SPEED LIGHT PEN SYSTEM

The LPS II is a true raster-scan light pen, the first ever made available for the Apple. This means that the instantaneous location of the Light Pen is determined at a full 60 cycle rate (sixty times each second), and that applications never before possible are made incredibly simple.

The Light Pen System comes with a standard software diskette in DOS 3.3 format. On the diskette is the PENTRAK driver and an entire selection of Applesoft application programs for the Light Pen, including four complete drawing programs which each take a different approach to Light Pen drawing. In addition, the interactive INSTALL utility customizes the PENTRAK driver for your particular hardware configuration, and calibrates the Light Pen to operate with your video monitor. The TYPEFACE LOADER program interactively loads supplementary Hi-Res typefaces for use in Hi-Res graphics annotation. OPART demonstrates full-screen animation on a scale never before seen on the Apple. The PENTRAK driver interfaces Applesoft programs to the Light Pen electronics. Using very simple commands, Applesoft programs can produce sophisticated interactive results with the LPS II.

The Sensor Diode in the point of the Light Pen is sensitive to the entire visible light spectrum. The Light Pen works equally well on monochrome white, green or amber monitors, and performs beautifully on color monitors and television sets of all kinds. RF modulators are no problem either. The Light Pen will function with any monitor you're currently using.

The LPS II requires an Apple II computer running Applesoft. This could be either an Apple II Plus, or an Apple II with a ROM or RAM card properly loaded. You must have at least one disk drive with DOS 3.3 proms installed in the disk controller and, of course, a video display monitor of any sort. Installation of the LPS II could not be easier; it simply plugs into an I/O slot of the Apple computer and is ready to go. While in the computer, the Light Pen does not interfere in any way with the normal operation of the system.

The LPS II includes complete user-oriented documentation taking you step-by-step from installation and calibration through basic operation and into advanced programming. The wide assortment of Application programs on the software diskette serve as additional user aids. Price is $349.00 + $5.00 from Gibson Laboratories, 406 Orange Blossom, Irvine, CA 92714.

BASIS 108, THE ALTERNATIVE

The BASIS 108 dual processor microcomputer system has features and capabilities not found in systems much more costly. But, more importantly, the system is totally compatible with the largest library of microcomputer software and plug-in peripherals available today. Its design and organization permit the BASIS 108 to be fully compatible with programs and peripherals originally developed for use with the Apple II™ microcomputer, including CP/M™-based applications software packages.

Standard features: 6502 and Z80 microprocessors; 64K RAM expandable to 128K; 2K system ROM, plus sockets for 10K additional user ROM; RGB and Composite Video; selectable 80 or 40 column text display; High Resolution Graphics: six colors, 280x192 or 280x160 with four lines of text; Color Graphics Mode: 15 colors 40x48 or 80x48; 40x40 or 80x40 with four lines of text; 8 bit parallel I/O; RS-232c serial I/O; Detached Keyboard: all standard keyboard functions, upper/lower case characters, numeric keypad, cursor control block, and 15 programmable special function keys; built-in mounting for two 5¼ inch floppy disk drives; six Apple II compatible slots for plug-in peripherals; game paddle I/O; built-in loudspeaker; two switched accessoy power outlets; mounting slots for up to six DP-25 I/O plugs. The price is $1949.00 from BASIS Inc., 23116 Summit Road, Los Gatos, CA 95030, (408) 353-1102, TWX: 910-590-8000.

A comprehensive review of the Basis 108 will appear in the August Call —A.P.P.L.E.

THE FRANKLIN ACE 100

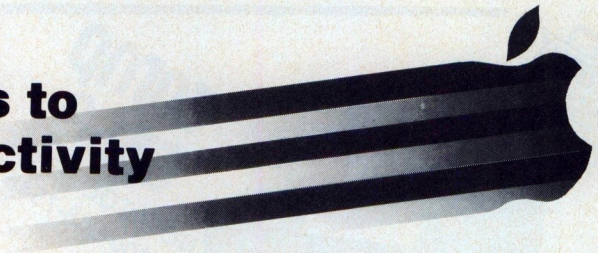
The Franklin ACE 100 is hardware and software compatible with the Apple® II. It is sweeter because it is more versatible, it includes 64K of RAM memory and—it costs less.

Any program that runs on the Apple II will run on the Franklin ACE 100. Any peripheral that works with Apple will work with ACE. The Franklin ACE 100 generates upper and lower case characters and includes a numeric pad, an alpha lock key and VisiCalc® keys.

Franklin ACE 100 features: Apple II compatible, 64K of RAM, Upper and lower case, Typewriter-style keyboard, Twelve key numeric pad, Alpha lock shift key, 50 watt power supply and built-in fan. The price is $1595.00.

For more information contact your computer dealer or contact Franklin Computer Corporation, 7030 Colonial Highway, Pennsauken, NJ 08109, (609) 488-1700.

S&H Software enables Apple][™ users to increase speed and productivity as much as 500%.



Universal Boot Initializer 4.0 works up to 500% faster than standard Apple DOS 3.3 . . . $69.95

UBI 4.0, with The DOS Enhancer,™ allows execution of BASIC and Binary programs up to 500% faster, depending on file length, and is completely compatible with standard Apple DOS 3.3 programs. For improved file management a new "FREE" command in DOS determines free space on disks.

UBI 4.0 breaks the language and time barriers between Apple][hardware and software by loading the RAM card with FPBASIC/INTBASIC (or your own program) in 1.7 seconds and by providing one-stage booting with DOS 3.3 or DOS 3.2.1 PROMS.

The UBI 4.0 package includes: training, utility, support and demo disks with complete documentation.

System requirements: 48K Apple][or][+, ROM/RAM card, DOS 3.3/DOS 3.2.1 and or more disk drives.

Amper-Sort/Merge (A-S/M) works up to 500% faster than even VisiCorp's VisiFile™ program...$52.95

The fastest "file clerk" you've ever met. Of all the sort utilities developed to manage Apple][data files, none does the job nearly so fast as A-S/M.

A-S/M can sort/merge from one to five unsorted files into a single file of up to 125K per disk.

Because sort routines can take up to 50% of computer running time in many business applications, you'll reap continuing benefits having the A-S/M "invisible speed demon" on your Apple][team -- saving 20 - 30 minutes a day of a human clerk's time spent "waiting" needlessly.

The A-S/M "speed demon" package includes: training disk and utility disks with complete documentation.

System requirements: 48K Apple][, ROM or RAM card, DOS 3.3 and one or more disk drives or 48K Apple][+, DOS 3.3 and one or more disk drives.

Available from your dealer. **Mail Order:** Send check to S&H Software, Box 5, Manvel, ND 58256. **Credit Cards:** Phone Cybertronics International Clearinghouse at 212-531-3089. (Overseas Airmail: Add $3.00 postage and handling.)

S&H Software

Box 5 Manvel ND 58256
(701) 696-2574

FILEMASTER

A DATA BASE, WORD PROCESSOR AND REPORT GENERATOR ALL COMBINED INTO ONE INTEGRATED SYSTEM.

WORD PROCESSOR

- Capitalization
  shiftlock access, displayed in inverse
- Indentation
- Insert-Delete
  characters of one or many
- Right-Left Margin Justification
  from 30 to 130 characters per line
- Variable Line Spacing
- Endless Applications
  form letters, diary, meeting notes, etc.
- Integrated with Report Generator
- Works with most any Printer
- All Instructions Visible
  instructions are displayed on the screen at all times

DATA BASE FEATURES

- Menu Driven
  instructions are visible on the screen at all times
- Up to 20 Fields per Record
- About 200 Records per File
- Up to 9 Files per Diskette
- Requires only One Disk Drive
- 48K Applesoft in ROM
- NO WRITE PROTECTION
  Use as many diskettes as you wish; make back-up copies or make modifications within the program.
- Works with most Printers
- Machine Language
  allows many of the routines to run several times faster for instantaneous response to commands while saving memory space.
- Easy Entry of Records
  enter records at any time, in any order.
- Instant Access to Records
  call all, some or one record to the screen instantly; move an entire block of records elsewhere in the file.
- Instantaneous Sort
  sorts through 1000 pieces of information per second, forwards or backwards, alpha-numerically or by number value.
- Instantaneous Search
  search a part or a whole file for a single character, a single word or a group of words.
- Divide and Merge
  divide a large file into two smaller files or merge two smaller files into one large file.
- Easy Update and Correction
  add, delete or rename files at any time; reorganize files or move an entire block of records within the file.
- Simple Mathematical Functions
  perform simple mathematical functions between fields such as payroll, checkbook, sales projections, etc.
- Field Modification
  add, delete, rename or reorganize fields at any time with ease.

REPORT GENERATOR

- Line Spacing
  single, double, triple, etc.
- Margin Spacing
- Tabulation
- Fillers
  dots, spaces, asterisks, etc.
- Fill in Preprinted Forms
- Create your own Forms
- Print Mailing Labels
  address and print form letters
- Telephone Lists
- Check Registers
- Shopping Lists
- Billing Statements
- Endless Applications
- Pre-Programmed Report Formats
  several to use at the push of a button
- Integrated with Data-Base and
- Word Processor

INCLUDES:

- 39 pages of Documentation
- Table of Contents
- Extensive Index
- One Program Diskette
- Instructions for setting up Checkbook Register and Telephone Directory
- 6½ x 9 inch Oxford Binder for handy storage of master diskette and documentation.

$149.00*

Michigan Residents add 4% sales tax.

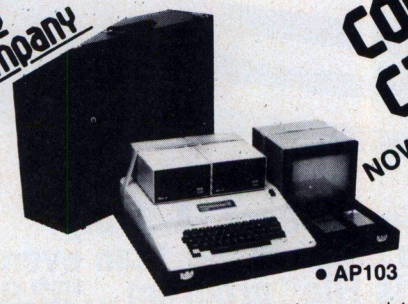
New Dealer Prices Available.

VISA & MC call 517-337-7423

Businessmaster Inc.

6443 COLEMAN RD. • E. LANSING, MI 48823

Computer Case Company



comp case

NOW TOLL FREE 800-848-7548

• AP103

Attaché style cases for carrying and protecting a complete computer set-up. Constructed of the highest quality luggage material with saddle stitching. Will accommodate equipment in a fully operational configuration along with manuals, working papers and disks. Never a need to remove equipment from case. Simply remove lid, connect power and operate.

• AP101	Apple II with Single Disk Drive	$109
• AP102	Apple II with Double Disk Drives	119
• AP103	Apple II, 9 inch Monitor & Double Drives	129
• AP104	Apple *III*, two additional Drives & Silentype	139
• AP105	12 inch monitor plus accessories	99
• P401	Paper Tiger 440/445/460	99
• P402	Centronics 730/737	89
• P403	Epson MX70 or MX80	89
• P404	Epson MX100	99
• P405	IDS 560 or Prism Printer	109
• CC80	Matching Attaché Case (5")	85
• CC90	Matching Attaché Case (3")	75
• CC91	Matching Accessory Case	95

computer case company

5650 INDIAN MOUND CT. COLUMBUS, OHIO 43213 (614) 868-9464

NOT COPY-PROTECTED!

by Charles R. Haight

DISKEDIT

BE THE

MASTER OF YOUR DISKETTES

with "the most versatile and user - friendly disk editing utility." *

TOTAL DISK ACCESS! VIEWS ANY DISK!

* DISKEDIT review in HARDCORE Computing

REVIVE CRASHED DISKS! RECOVER DELETED FILES!

Insert or remove illegal characters.
Write flashing and inverse titles.
Hide or disguise file names.
Customize the CATALOG

Source Code and Fully-Commented Listing included on disk.

ONLY **$25** U.S.A.

SOFTKEY PUBLISHING DEPT 222
P.O. Box 44549
Tacoma, WA 98444

for 48K Apple II PLUS with Applesoft in ROM.
NO Credit Cards. Dealer Inquiries Invited.

DOS PATCH FOR 80 COLUMN CARDS

Glen Bredon

A number of copies of BIG MAC (LC) were sent out without a DOS modification which was designed to prevent interference with 80 column cards resulting from inverse and/or flashing characters in the catalog. They also lacked a patch which enabled the control C catalog abort feature at the catalog pause point. Moreover, the disks were masters rather than slaves, so that disk boots would write over source files.

It is a simple matter to re-establish these patches on another disk. To do this, first boot your 3.3 system master disk so that a clean DOS is in memory. Type new to get rid of the Hello program. Then enter the monitor by CALL -151 and carefully enter the following data:

```
9E42:34
AE3A:F2 B6
AE1C:DD B6
B6DD:30 0A 46 32 46 32 29 3F
69 1F 49 E0 20 ED FD A9 FF 85 32
60 00
B6F2:20 0C FD C9 83 F0 01 60 4C
7F B3
```

Next type B6DDL to list the program just entered and check it with care against the listing supplied here. If it differs, it is wise to start over at the beginning.

Finally, insert a blank disk and type INIT BIG MAC. When the INIT is finished you should DELETE BIG MAC. Then you can use FID to transfer the files you wish to have on the new disk. CONFIGURE ASM could also be used to transfer the files BIG MAC and ASM.OBJ.

```
B6DD:  30  0A
B6DF:  46  32
B6E1:  46  32
B6E3:  29  3F
B6E5:  69  1F
B6E7:  49  E0
B6E9:  20  ED  FD
B6EC:  A9  FF
B6EE:  85  32
B6F0:  60

B6F1:  00

B6F2:  20  0C  FD
B6F5:  C9  83
B6F7:  F0  01
B6F9:  60
B6FA:  4C  7F  B3
  --End assembly--
```

To change the DOS on the original disk, or a copy of it, you can use a copy program that allows you to copy individual tracks, and copy just tracks 0,1,2 from the new disk you have just fixed to the original. If you do not have such a copy program, you might make it one of your projects to write one of your own. It is really not hard and is just a matter of learning how to use the RWTS routine in DOS. This is the way to learn how to do things of this sort. Just remember to thoroughly test out such a program on scratch disks before you try it on anything important.

These two patches can be used on other disks and are not specifically restricted to BIG MAC. People with 80-column cards might consider using them as part of their standard DOS. The modification at $9E42 is the change required to use a binary program as the boot program and is not related to the other patches.

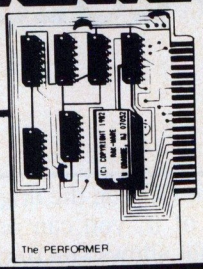
It should be noted that BIG MAC makes some other patches to DOS for its own use. Thus it is important that you not try to patch any other locations in DOS for your BIG MAC disks.

```
1     ********************************
2     *                              *
3     *   DOS patches for CATALOG    *
4     *                              *
5     *   For 48K DOS 3.3 only!!!    *
6     *                              *
7     *     G. Bredon    1/1/82      *
8     *                              *
9     ********************************
10    *                              *
11    *    Also make:                *
12    *                              *
13    *    AE3A: F2 B6               *
14    *    AE1C: DD B6               *
15    *                              *
16    *    Modification of the       *
17    * addresses of these routines  *
18    * may interfere with proper    *
19    * operation of BIG MAC!!!      *
20    *                              *
21    ********************************
22
23
24    INVFLG    =     $32
25    COUT      =     $FDED
26    RDKEY     =     $FD0C
27    FMDONE    =     $B37F       ;Exit routine from file mngr
28
29              ORG   $B6DD       ;Ok only for DOS 3.3
30
31    ********************************
32
33              BMI   GCOUT       ;Treat - chr as usual
34              LSR   INVFLG      ;Kill 2 high bits of INVFLG
35              LSR   INVFLG
36              AND   #%00111111  ;Inv & flash -> uc
37              ADC   #$1F        ;Carry is set
38              EOR   #$E0
39    GCOUT     JSR   COUT        ;Let COUT handle inverse
40              LDA   #$FF        ;Reset INVFLG to normal
41              STA   INVFLG
42              RTS
43
44              BRK
45
46              JSR   RDKEY       ;This must ORG $B6F2
47              CMP   #$83        ;Control C at catalog pause?
48              BEQ   ABORT       ;Skip rest if so
49              RTS               ;Otherwise send rest of cat
50    ABORT     JMP   FMDONE      ;Aborts catalog routine
   32 bytes     Errors: 0
```

HARDWARE

APPLE

201 - 839-3478
Dealer and Distributor Inquiries Invited.

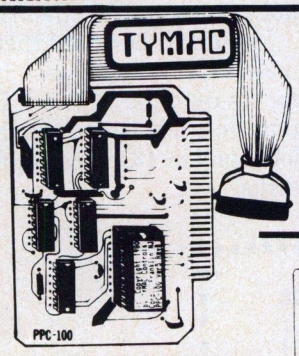


THE PERFORMER PRINTER

FORMATTER BOARD for Epson, OKI, NEC 8023, CITOH 8510 provides resident screen dump and print formatting in firmware. Plugs into Apple slot and easily accessed through PR# command — Use with standard printer cards. $49.00 specify printer.

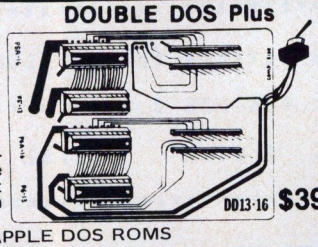
THE MIRROR FIRMWARE FOR NOVATION APPLE CAT II®
The Data Communication Handler ROM Emulates syntax of another popular Apple Modem product with improvements. Plugs directly on Apple CAT II Board. Supports Videx and Smarterm 80 column cards, touch tone and rotary dial, remote terminal, voice toggle, easy printer access and much more.
List $39.00 — Introductory Price $29.00

PARALLEL PRINTER CARD



A Universal Centronics type parallel printer board complete with cable and connector. This unique board allows you to turn on and off the high bit so that you can access additional features in many printers. Use with EPSON, C.ITOH, ANADEX, STARWRITER, NEC, OKI and other with standard Centronics configuration.
$139.00

DOUBLE DOS Plus



DOUBLE DOS Plus — a piggyback board that plugs into the disk-controller card so that you can switch select between DOS 3.2 and DOS 3.3
DOUBLE DOS Plus requires APPLE DOS ROMS
$39

SOFTWARE

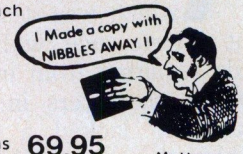
APPLE 48K

201 - 839-3478

NIBBLES AWAY II

AGAIN! Ahead of all others.

- **AUTO-LOAD PARAMETERS** . . . Free's the user from having to Manually Key in Param values used with the more popular software packages available for the Apple II.
- **EXPANDED USER MANUAL** . . Incorporates new Tutorials for all levels of expertise; Beginners Flowchart for 'where do I begin' to 'Advanced Disk Analysis' is included.
- **TRACK/SECTOR EDITOR** . . . An all new Track/Sector Editor, including the following features: Read, Write, Insert, Delete Search, and impressive Print capabilities!
- **DISK DIAGNOSTICS** . . . Checks such things as: Drive Speed, Diskette Media Reliability, and Erasing Diskettes.
- **HIGHEST RATED**. . .Best back up Program in Softalk Poll (Rated 8.25 out of 10).
- **CONTINUAL UPDATES** . . . Available from Computer Applications and new listings on the source.



69.95

Mr. Lister

Customer Contact Profiler & Mailer

A Super Mail List Plus more up to 1000 Entries on single 3.3 Disk (only 1 Drive required) 2 second access time to any name full sort capabilities Dual Index Modes supports new 9 digit Zip. Easy to follow manual Not Copy Protected 4 user defined tables with 26 sort selections per table Beta tested for 6 months user defined label generation.
INTRODUCTORY PRICE ~~$139~~. $99.00 Dealer & Dist. Inquiries Invited.

Super PIX HIRES SCREEN DUMP

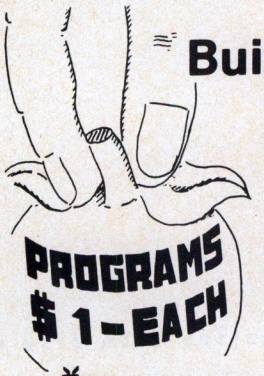
The Software package that will allow your printer to dump page 1 or page 2 of the Apple Hires screen horizontally or vertically. Use with EPSON® MX-80 with or without GRAFTRAX® Roms, MX-70 – OKI® Microline 80, 82, 83, 82A, 83A – C. ITOH® 8510 and NEC 8023A. Requires Tymac Parallel Printer Board PPC-100, ... $24.95

THE APPLE CARD—Two sided 100% plastic reference card Loaded with information of interest to all Apple owners $3.98

MasterCard VISA

MICRO-WARE DIST. INC.
P.O. BOX 113
POMPTON PLAINS, N.J.
07444

Dealer and Distributor Inquiries Invited.



Build and Enhance

your Apple * software library at affordable prices.
For the FIRST time — PACKED disks with 60 outstanding programs.
Your Apple * needs programs! At last you have an opportunity to use your Apple* to its fullest capacity!

APPLEWARE INC. offers on each disk an extensive variety of programs including:

Business	• Printer	• Graphics
Games	• Text	• Music
Utilities	• Data Base	• Finance
Science	• Engineering	• and many

others with 3.3 DOS for easy demuffining.

First 3 disks available at $59.95 each
Order all 3 for a free bonus disk.
Call now toll free: 1—800-327-8664
Fla. Residents: 305—584-7004
Please add $3 for postage & handling.

6400 Hayes St.
Hollywood, FL 33024

VISA master charge

*TM Apple is a registered trademark of Apple Computer Co.

You've Seen the Rest Now Get the Best

GLOBAL Program Line Editor

Global PLE has all these fine new features to further ease your editing tasks:

- Global edit, search and replacement
- Compatible with 80 column cards
- Typeahead buffer
- 1152 bytes of Escape functions definable from keyboard and nestable
- Page list mode
- Completely transparent to user
- Loads in regular RAM or Language card.

Plus all of the powerful standard PLE commands and features

$38.50

A.P.P.L.E. Orders
304 Main Ave. S., Suite 300
Renton, WA 98055
(206) 271-4514

Washington state residents add 6.5% sales tax.
All foreign orders (including Canada) will require additional postage and shipping.
Shipping wt. 3.0 oz.

A CALL —A.P.P.L.E. UTILITY:

BINARY TO TEXT CONVERTER

Val J. Golding with Dave Lingwood

LEAVE it to Lingwood is a favorite saying around our neck of the woods. Dave, needing to transmit a binary file to us via a modem and the phone, devised the basis for this short utility. While one can, with a bit of determined effort, transmit binary files over the phone, it is usually far simpler to handle a text file, the file type most terminal programs expect to find.

Apple's monitor has a "hex dump" routine that prints a memory address in hex followed by a dash and eight bytes of data, each separated by a space. This is what you see when you enter monitor and type, for example, 300.308.

Even though Dave's gem did its job, when we got our hands on it, we spotted an additional use to which it could be put, to save a binary file as text, then EXEC the file into memory. Only three things need be done, we replace the monitor's dash with a colon, add an opening line to the text-file to CALL -151, and add a closing line to return to BASIC. Since these are optional features, they are shown in the BASIC listing only as REMs. Should you wish to use them, just remove the REM from the statement.

Another innovative feature we would like to suggest our readers study and perhaps improve upon, is the method of loading the binary code in memory. This converts the decimal starting address to HEX characters that the Lam routine can use, limited to one byte. Thus it could be used only for short programs using memory up to $CFF.

The BASIC program is set up to place the machine code in memory using the S. H. Lam routine, so all you need do is type it in and fire away. The Assembly listing is provided for those that wish to explore the program more fully.

```
0300: AD 72 AA
0303: 85 00
0305: 18
0306: 6D 60 AA
0309: 48
030A: 85 02
030C: AD 73 AA
030F: 85 01
0311: 6D 61 AA
0314: 85 03
0316: 68
0317: D0 02
0319: C6 03
031B: C6 02
031D: 38
031E: B0 06

0320: A5 00
0322: 29 07
0324: D0 13

0326: A4 01
0328: A6 00
032A: A9 8D
032C: 20 ED FD
032F: 20 40 F9
0332: A0 00
0334: A9 BA
0336: 20 ED FD

0339: A9 A0
```

```
2     ********************************
3     *                              *
4     * Routine to save binary data  *
5     * and addresses to a text file *
6     *                              *
7     * Stolen from Apple's monitor  *
8     * by D. Lingwoood/V. Golding   *
9     *                              *
10    * This code fully relocatable  *
11    *                              *
12    * Call -A.P.P.L.E. : July 1982 *
13    *                              *
14    ********************************
15    *
16    STADR    =    $00
17    NDADR    =    $02
18    BLEN     =    $AA60
19    BSTART   =    $AA72
20    PRNTYX   =    $F940
21    PRBYTE   =    $FDDA
22    COUT     =    $FDED
23    *
24    *
25              ORG  $300
26    *
27    *
28              LDA  BSTART      ;Get start addr from DOS
29              STA  STADR       ;Save it
30              CLC
31              ADC  BLEN        ;Add the length
32              PHA              ;Check later
33              STA  NDADR       ;and save it
34              LDA  BSTART+1    ;Now do same for hi byte
35              STA  STADR+1
36              ADC  BLEN+1
37              STA  NDADR+1
38              PLA
39              BNE  LOWONLY
40              DEC  NDADR+1
41    LOWONLY   DEC  NDADR       ;Fix DOS goof
42              SEC
43              BCS  PRA1        ;Unconditional branch
44    *
45    MOD8CHK   LDA  STADR
46              AND  #$07
47              BNE  DATAOUT
48    *
49    PRA1      LDY  STADR+1     ;Output routine starts here
50              LDX  STADR
51    PRYX2     LDA  #$8D
52              JSR  COUT        ;Print a [C/R],
53              JSR  PRNTYX      ;memory address
54              LDY  #$00
55              LDA  #":"        ;and a colon
56              JSR  COUT
57    *
58    DATAOUT   LDA  #"          " ;Space
```

BINARY TO TEXT CONVERTER

```
0338:  20 ED FD            JSR  COUT         ;Output the space
033E:  20 DA FD            JSR  PRBYTE       ;Output byte in hex
0341:  A5 00      NXTA1    LDA  STADR        ;Increment 2 byte start addr
0343:  C5 02               CMP  NDADR
0345:  A5 01               LDA  STADR+1      ;and compare to 2 byte end addr
0347:  E5 03               SBC  NDADR+1
0349:  E6 00               INC  STADR        ;(Carry set if >=)
034B:  D0 02               BNE  MOD8
034D:  E6 01               INC  STADR+1
034F:  90 CF      MOD8     BCC  MOD8CHK      ;Ready for next addr?
0351:  60                  RTS               ;Back to BASIC
--End assembly--          82 bytes           Errors: 0
```

```
]LIST

10  REM

"        BINARY TO TEXT SAVE

by Dave Lingwood & Val J Golding

  Call -A.P.P.L.E. : July, 1982

100  HOME : VTAB 4: PRINT "      H
     EX LIST TO DISK": PRINT
110  GOSUB 800
120 D$ =  CHR$ (13) +  CHR$ (4)
130  INPUT "FILE TO COPY ? ";F1$:
      INPUT "SOURCE DRIVE (1/2) ?
     ";D1$: ON  NOT  VAL (D1$) OR
      VAL (D1$) > 2 GOTO 130: IF
     F1$ = "" THEN  PRINT D$"CATA
     LOG": GOTO 130
140  PRINT D$"BLOAD"F1$",D"D1$
150  INPUT "TARGET DRIVE (1/2) ?
     ";D2$: INPUT "FILENAME TO SA
     VE ";F2$: ON  NOT  VAL (D2$)
      OR  VAL (D2$) > 2 GOTO 150:
      IF F2$ = "" THEN 150
160  PRINT D$"MONCIO"D$"OPEN"F2$"
     ,D"D2$D$"WRITE"F2$
170  REM  PRINT"CALL-151"
180  CALL BP * 256: REM
M/L start address

190  PRINT
200  REM PRINT"3D0G"
210  PRINT D$"CLOSE": END : REM

800 X$ = "0123456789ABCDEF"
900 BP =  PEEK (110) + 2: FOR I =
     1 TO BP:H$ =  MID$ (X$,I + 1
     ,1): NEXT : REM

Find free space pointer, convert
address to hex, stuff in string

910 H$ = H$ + "00:AD 72 AA 85 0 1
     8 6D 60 AA 48 85 2 AD 73 AA
     85 1 6D 61 AA 85 3 68 D0 2 C
     6 3 C6 2 38 B0 6 A5 0 29 7 D
     0 13 A4 1 A6 0 A9 8D 20 ED F
     D 20 40 F9 A0 0 A9 BA 20 ED
     FD A9 A0 "
920 H$ = H$ + "20 ED FD B1 0 20 D
     A FD A5 0 C5 2 A5 1 E5 3 E6
     0 D0 2 E6 1 90 CD 60 ND9C6G"

930  FOR I = 1 TO  LEN (H$): POKE
     511 + I, ASC ( MID$ (H$,I,1)
     ) + 128: NEXT : POKE 72,0: CALL
      - 144: RETURN : REM

Stuff machine code into memory
using Lam routine
```


$$

CLOSEOUT SPECIAL

The WOZPAK][	**$20.00**
WOZPAK DISKETTE	**17.50**
Total	~~**$37.50**~~ $20.00

While Supply Lasts,* you can save ~~$12.50~~ $17.50 and get both for just ~~$25.00~~ $20.00.

The WOZPAK][is the source for Integer Basic ROM technical information, containing articles on the Integer Floating point routines, Sweet 16, Integer line number and symbol cross reference programs and related information.

A.P.P.L.E. Orders
304 Main Ave. S., Suite 300
Renton, WA 98055
(206) 271-4514

Washington state residents add 6.5% sales tax.

All foreign orders (incl. Canada) will require additional postage and shipping.

*Less than 200 copies remain

Shipping weight 13.02 oz.

A.P.P.L.E. Members only

$$

DOUBLE your DISKETTES

The only reasons your APPLE II cannot use the back side of your diskette are:

1. There is no notch.
2. The diskette manufacturer either did not test the back side, or worse, put the flawed side to the back.

A pair of diagonal cutters will solve problem number 1.

DISK PREP

will solve problem number 2.

DISK PREP formats and tests your disk. Sectors with flaws are left so they cannot be used. Your disk is left ready to boot, complete with a flaw report program saved on it.

$25.00

sympathetic software
9531 Telhan Drive
Huntington Beach, Ca. 92646
DOS 3.3 and DOS 3.2.1
California residents add $1.50 tax.

MOVING BIG MAC SYMBOL TABLE

Thomas G. Williams

THE Symbol Symon symbolic debugger is a major improvement over the capability included in the Apple][monitor. Unfortunately, it does not know about the Big Mac.LC assembler and thus can not save or use the symbol table produced by Big Mac.LC. Symbol Symon can be used without a symbol table, but you lose the capability to refer to locations by name instead of address. Fortunately, the Big Mac.LC symbol table is in the same format as the Big Mac.STD symbol table, which Symbol Symon can use. If the Big Mac.STD symbol table is moved to where Big Mac.STD would have put it, and if certain memory locations are set to values that Big Mac.STD would have set, Symbol Symon can find and use the symbol table.

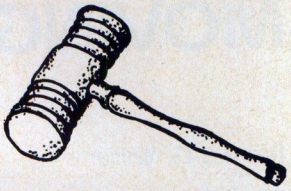
When the Symbol Symon "Save Symbol Table" option is selected, Symbol Symon checks locations $8BC–$8C0 for the sequence $02,$09, $07,$20,$0D ("BIG M" in inverse video). If it finds this sequence, it assumes that Big Mac.STD was used and takes the two byte datum stored at $80D,$80E as the address of the last byte of the assembler source file. Big Mac.STD starts the symbol table immediately after the source file.

Big Mac.LC puts the symbol table on the language card in bank one of $D000–$DFFF. This program is for use with assembly language programs where all of the symbol table will fit into this space. It moves the data from bank one of $D000–$DFFF to $7000–$7FFF. It then puts the proper values on $8BC–$8C0 and sets $80D–$80E to $6FFF. Thus the Symbol Symon "Save Symbol Table" option will think Big Mac.STD was used and save the data starting at $7000. This program always moves 4096 bytes; it makes no attempt to identify the end of the symbol table. This works fine because Symbol Symon can identify the end of the table and will not save extraneous data.

To use the program, BRUN it after the assembly is complete but before Symbol Symon is started. As given here, it runs in page 3 of memory. Neither Big Mac.LC nor Symbol Symon uses page 3, so if the program you are debugging does not use page 3, this program will not be disturbed, and can be executed by a CALL 768 after exit from Big Mac.LC.

```
:ASM
                        ********************************
                        *                              *
                        * MOVE BIG MAC.LC SYMBOL TABLE *
                        *       by  Tom Williams       *
                        *                              *
                        * Call -A.P.P.L.E. : July 1982 *
                        *                              *
                        * Move Big Mac.LC Symbol table *
                        * to where it would be if      *
                        * assembled by Big Mac.STD so  *
                        * SYMBOL SYMON can find it.    *
                        *                              *
                        * Move bank 1 $D000-DFFF to    *
                        * $7000-$7FFF. Put $6FFF (end  *
                        * of source) in $80D-80E. Put  *
                        * proper values in $8BC-$8C1.  *
                        *                              *
                        ********************************
                        *
                        * Equates
                        *
                        A1L       EQU  $3C
                        A1H       EQU  $3D
                        A4L       EQU  $42
                        A4H       EQU  $43
                        ENDSRC    EQU  $80D        ;End of source
                        CHKLOC    EQU  $8BC        ;Loc'ns checked by SS
                        *
                        *
                                  ORG  $300
                        *
                        * Move symbol table
                        *
0300: 2C 88 C0          START     BIT  $C088       ;Select RAM read, bk 1
0303: A9 00                       LDA  #0          ;Set up src, dest ptrs
0305: 85 3C                       STA  A1L
0307: 85 42                       STA  A4L
0309: A9 D0                       LDA  #$D0        ;Source
030B: 85 3D                       STA  A1H
030D: A9 70                       LDA  #$70        ;Destination
030F: 85 43                       STA  A4H
0311: A0 00                       LDY  #0          ;Clr Y for indir addr
0313: B1 3C             LOOP      LDA  (A1L),Y     ;Get Datum
0315: 91 42                       STA  (A4L),Y     ;Store Datum
0317: E6 3C                       INC  A1L         ;Increment pointers
0319: E6 42                       INC  A4L
031B: D0 F6                       BNE  LOOP
031D: E6 3D                       INC  A1H         ;Must increment hi byte
031F: E6 43                       INC  A4H
0321: A5 3D                       LDA  A1H         ;Check if done
0323: C9 E0                       CMP  #$E0
0325: D0 EC                       BNE  LOOP
                        *
                        * Move finished, set end of source pointer
                        *
0327: A9 FF                       LDA  #$FF
0329: 8D 0D 08                    STA  ENDSRC
032C: A9 6F                       LDA  #$6F
032E: 8D 0E 08                    STA  ENDSRC+1
                        *
                        * Set up locations checked by SYMBOL SYMON
                        *
0331: A2 04                       LDX  #4
0333: BD 40 03          LOOP1     LDA  CHKDATA,X
0336: 9D BC 08                    STA  CHKLOC,X
0339: CA                          DEX
033A: 10 F7                       BPL  LOOP1
                        *
                        * Reset to ROM read
                        *
033C: 2C 81 C0                    BIT  $C081
033F: 60                          RTS
                        *
                        * Data checked by SYMBOL SYMON
                        *
0340: 02 09 07
0343: 0D                CHKDATA   HEX  02,09,07,0D
 --End assembly--          68 bytes     Errors: 0
```

A.P.P.L.E. MEETING MINUTES



Michael Thyng/guest secretary

May 18, 1982
Computer Empire

VICE President Bob Huelsdonk called the meeting together at 7:33 PM, sixty five members were present.

Old Business:

A.P.P.L.E. office hours are extended to 7:00 PM on Thursday evenings. An informal show of hands indicated that 95% of the members currently receive *Softtalk*. No further news about the possibility of A.P.P.L.E. sponsoring subscriptions. It is still under investigation. There were some questions on the results of the user survey. Some asked when it would be published. David Lingwood was unavailable for comment.

New Business: None.

Committee Reports:

Mike Thyng reported that there were many volunteers for the Pacific Science Center and A.P.P.L.E. co-sponsored, first annual Apple Computer Festival. He said there were members coming in from Seattle, Tacoma, Bend, Eugene, Bremerton, and Washougal to participate. There were visitors coming from Canada. Others were encouraged to participate.

Other Reports:

Val Golding gave a first-hand account of his impressions of the Boston Applefest. He said that Boston had improved its facilities from a four aisle show last year to a fourteen aisle extravaganza this year. He saw only a few new products. Among them, a new Z-80 card which features on-board 64K RAM and your choice of a four or six mhz chip. Val thought it ran faster than Microsoft's Z-80.

The star of the show seemed to be the first ever high resolution light pen. This comes with a $349.00 plug-in card. The Franklin Ace 100 - a dual processor, Apple similar computer - was displayed. XEBEC demonstrated an Apple compatible hard disk. The 5¼ inch five megabyte storage device sells for about $1300.00

A.P.P.L.E. picked up about 85 new members. We adjourned the formal business portion of the meeting at 7:50.

Speaker:

Tom Benett of the University of Washington Academic Computer Center spoke to us about communicating over phone lines to a large host computer. Tom is the Terminal Services Coordinator at the "U." He told us that to talk to the university's computer - which is a mainframe CDC Cyberg 170-750, a 60 bit word machine - you needed three things.

1) Hardware interface; the RS232-C
2) A Modem
3) Software to let your Apple emulate a terminal

A modem translates electronic pulses to and from the Apple into analog signals for transmission over the phone lines. The software you buy for emulating a terminal comes in two major catagories. dumb terminal and smart terminal.

A dumb terminal gives you the ability to talk to the host computer, to load programs, to read data that the host has available and to enter data remotely through your keyboard. A smart terminal lets you do all that, plus it allows you to transfer data (text) files to or from your disk drive from or to the host computer. Tom described an abundance of data available for the users of the University of Washington computer center.

Rates are very low for non-commercial users. To sign up to use this system or to get more information, you may call Tom at (206) 543-5970. Tom left us with a thought that microcomputer owners should carry into the '80's. "The best way to compute is to let the microcomputer do what it does best and let the mainframes do what they do best."

Q & A

There were questions about the rumors of new Apple computers, why a system would not boot up, comments about the large capacity boards that simulate disk drives without diskettes, and a comment that one piece of software called *Crunch* correctly removed REM statements, but then went ahead and removed some semi-colons too.

MICRO MANTIC COMPUTER™

TAKE CONTROL OF YOUR APPLE® DISKS

YOU CAN NOW HAVE UNEQUALLED ACCESS AND CONTROL OF YOUR DISKS WITH THE ULTIMATE DISK UTILITY PACKAGE

DISKOVERY™

UTILITIES INCLUDED:

TRACK/SECTOR EDITOR

The most VERSATILE and USEFUL sector editor available! Among the more than 70 commands (with more on the way!) are:

- Simultaneous operation in both 13 and 16 sector formats.
- Compatible with 35, 40, 70, and 80 track drives.
- Compatible with CP/M® and PASCAL™
- 128 sector holding area with fully automatic index.
- Automatic index keeps track of all 128 holding blocks' origin information including slot, drive, track, and sector.
- Search for a string of up to 255 characters within a file or sequentially by selected tracks.
- Reconstruct a VTOC (Volume Table Of Contents).
- Reconstruct catalog sectors.
- Verify all used sectors on a disk.
- Initialize any one track or range of tracks.
- Load or Save up to 8 tracks at one time.
- Load or Save files of up to 128 sectors long at one time.
- Compare any held sector or range of sectors to any other sector or range of sectors.
- Edit sectors in Hex or ASCII. When in ASCII, characters may be entered as inverse, flashing, control, or normal.
- 3 Hardcopy formats with 3 selectable line lengths. Any printer initialization string can be sent to output slot.
- Locate all track/sector lists on a disk.
- Ability to override checksum check to read bad sectors.
- Ability to mask display to show inverse and flashing as normal mode characters.

Constant on screen indication of active slot and drive, tracks on the disk, disk volume #, present track and sector, mask condition, DOS type in use, last hold block used, and number of hold blocks currently not being used for storage.

DRIVE SPEED TEST

High accuracy disk drive speed testing program. Displays drive rotation speed with an accuracy of +/- 20 microseconds.

CATALOG MANAGEMENT TOOLS

Alphabetize the catalog by file names or by file names within file types.
Purge all deleted files from catalog.
Patch DOS to show deleted files during a catalog command.
Undelete a file in the catalog - checks for file integrity.

MISCELLANEOUS UTILITIES

Initialize disk without DOS. Increases usable disk space.
Erase disks to a totally unformatted state.
DOS patches to allow BRUN or EXEC execution of boot file.
Setup program to allow use of lowercase adapters and disk drives other than APPLE DISK II.

Extensive manual packed with lots of valuable information.
Disk supplied comes unprotected to allow easy customizing for special user applications and user patches.
Program updates GUARANTEED FOR 5 YEARS. Planned updates include macros for the editor, rename DOS commands, text file editor for sequential and random files, and many more.

INTRODUCTORY OFFER

$59.95

COD, Check, or Money Order
Requires Apple II/II+ with 48K and disk drive. Specify 3.2 or 3.3 format.

Micro Mantic Computer
541 NE McWilliams Rd
Bremerton, WA 98310
Call now (206) 373-9231

APPLE is a registered trademark of Apple Computer, Inc. CP/M is a registered trademark of Digital Research, Inc.
PASCAL is a trademark of U.C.S.D. Regents.

THEY CAME OUT OF THE BLUE OF THE WESTERN SKY

and it's up to you to stop them!

REAR GUARD

FEATURING ARCADE ACTION GRAPHICS AND SOUND

You know those alien residents who look like blue jello and smell like a cow pasture on a warm day? Well, they're up in arms about something and have declared war! Their weapons are Cyborg ships with their intelligent parts suspended inside. These Cyborgs pursue like crazed bats to try and kamikaze your crew pods. You're armed with an unlimited supply of Guided Energy Darts, but even when you blast 'em, their doomsday missiles come right at you! Good luck. (You'll need it!)

REAR GUARD features many extras which are usually found only on coin-operated video arcade machines. These include advanced play levels, running high score (with high score "clear" option), two player option, brilliant horizontal scrolling, and great sound effects.

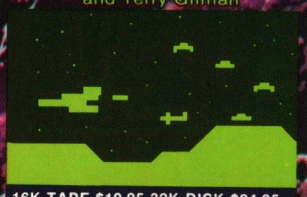
APPLE VERSION
by John Anderson

48K DISK $29.95

ATARI VERSION
by Neil Larimer
Assisted by Sparky Starks

16K TAPE $19.95 24K DISK $24.95

TRS-80 VERSION
by Wayne Westmoreland
and Terry Gilman



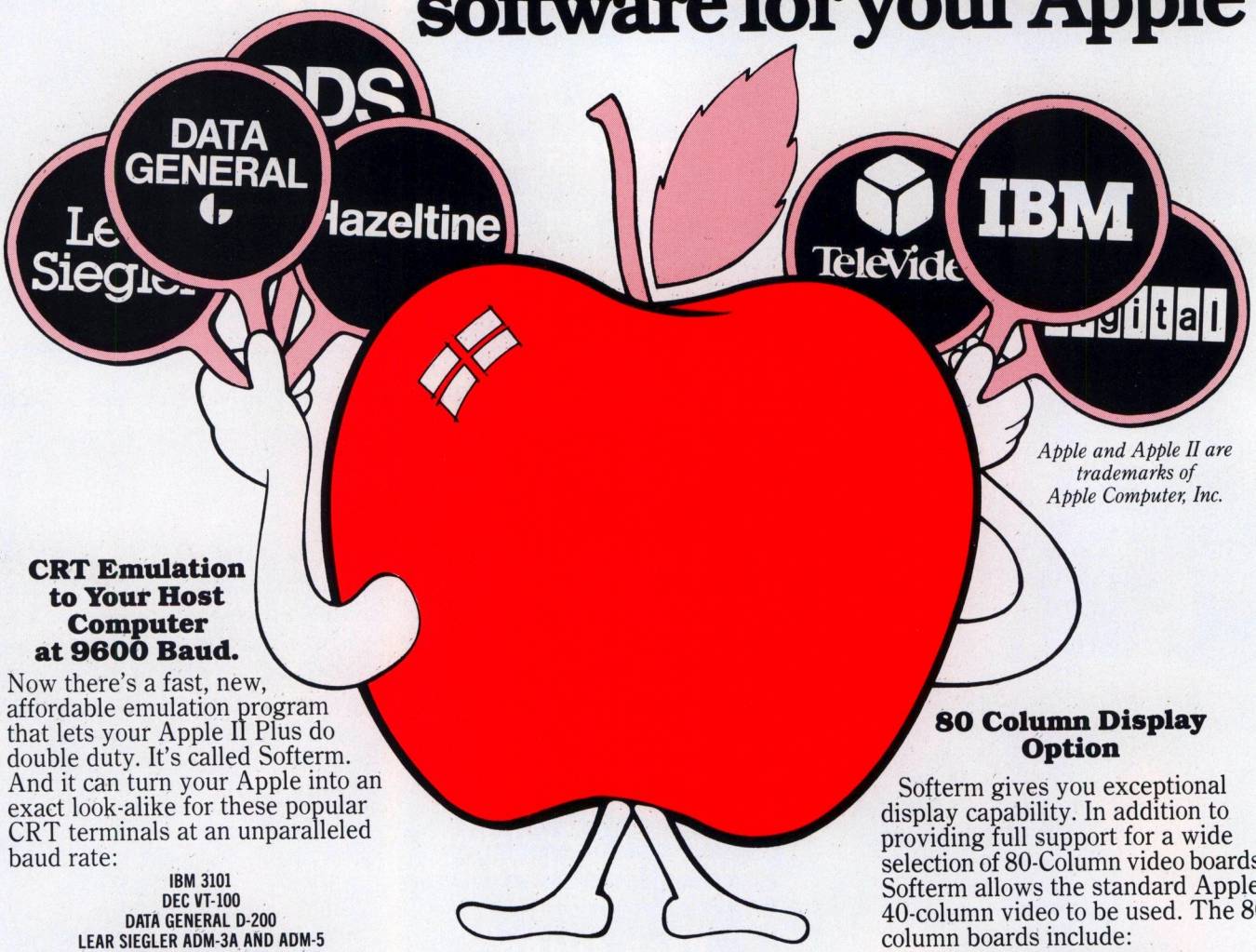
16K TAPE $19.95 32K DISK $24.95

TO ORDER!

Adventure INTERNATIONAL
A DIVISION OF SCOTT ADAMS, INC.

BOX 3435, LONGWOOD FL 32750
(305) 862-6917 (QUESTIONS)
ORDER FROM YOUR FAVORITE DEALER
or CALL TOLL FREE (800) 327-7172
(ORDERS ONLY PLEASE)

Finally. High-speed CRT look-alike software for your Apple



Apple and Apple II are trademarks of Apple Computer, Inc.

CRT Emulation to Your Host Computer at 9600 Baud.

Now there's a fast, new, affordable emulation program that lets your Apple II Plus do double duty. It's called Softerm. And it can turn your Apple into an exact look-alike for these popular CRT terminals at an unparalleled baud rate:

IBM 3101
DEC VT-100
DATA GENERAL D-200
LEAR SIEGLER ADM-3A AND ADM-5
HAZELTINE 1400 AND 1500 SERIES
ADDS REGENT SERIES
TELEVIDEO 900 SERIES

Softerm gives you the security and convenience of a personal computer *plus* the benefits of an on-line terminal. You can even transfer files simply, quickly, and safely. With Softerm you can access your mainframe, talk to your timesharing service, or communicate with another Apple computer.

Simple to use

Softerm operates on any Apple II Plus with 48K memory and Apple II disk drive. For a connection to another computer directly or through a communications modem, you'll need an asynchronous serial interface board. Here are the boards Softerm supports:

APPLE COMMUNICATIONS CARD
APPLE HIGH SPEED SERIAL INTERFACE
CALIFORNIA COMPUTER SYSTEMS 7710
MOUNTAIN COMPUTER CPS MULTIFUNCTION CARD
SSM MICROCOMPUTER PRODUCTS SIO, AIO, AIO II™
HAYES MICROCOMPUTER PRODUCTS—
MICROMODEM II™, SMARTMODEM™
NOVATION APPLE-CAT™
BIT 3 COMPUTER DUAL—COMM PLUS™

Flexible file transfer

Softerm makes it simple to transfer text, program, and binary files from one computer to another. You'll enjoy a variety of modes to satisfy your host computer's requirements, including character, block, and the intelligent Softrans™ mode which guarantees error-free transmission and reception.

Other mode selections offered include manual or unattended operation, auto-dial and auto-answer, adjustable character and block delays, character echo wait, handshake sequencing, unattended multiple file transmit queueing, receive to print, and many others.

80 Column Display Option

Softerm gives you exceptional display capability. In addition to providing full support for a wide selection of 80-Column video boards, Softerm allows the standard Apple 40-column video to be used. The 80 column boards include:

ADVANCED LOGIC SYSTEMS SMARTERM™
VIDEX VIDEOTERM™
M & R ENTERPRISES SUP-R-TERM™
BIT 3 COMPUTER FULL VIEW 80™
VISTA COMPUTER VISION 80™
WESPER MICRO SYSTEMS WIZARD 80™
COMPUTER STOP OMNIVISION™

Most advanced emulator available.

Softerm is written entirely in 6502 assembly language. You simply can't buy a more sophisticated package or one that's easier to use. Softerm is the most advanced intelligent communications software on the market.

Order Softerm™ now.

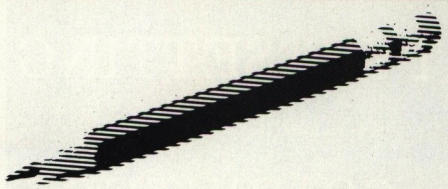
The sooner you can put Softerm to work for you the sooner you'll put the power of other computers at your fingertips. So order your Softerm program today and discover a world of new uses for your Apple. Send $150 to Softronics, Inc.

SOFTRONICS

6626 PRINCE EDWARD PLACE • MEMPHIS, TN 38119 • (901) 755-5006

MasterCard
VISA

WRITE -A.P.P.L.E.



The Readers

Dear Val,

HAVING recently programmed a C.A.I. package containing several educational games, I noticed that one of the games exhibited a flaw when invoking the random (RND) function. The game was a variation on the wordgame "Hangman," and the mystery words were chosen from a data array "randomly." I use this word metaphorically only, as it seems that the choice was not so random at all. After trying out the program repeatedly—it seemed that every time you began to play the game anew, the same words tended to come up first.

Upon complaining of this to a friend who programs for a C.A.I. company, he told me there was a bug in the Applesoft RND function and that there was a simple fix for it! (namely POKE 205,255) I happily tried out the fix only to see the problem persist!

I narrowed it down to this observation: when using the RND function from a cold start (power up situation), the Apple will always give you the same sequence of random numbers. Later I learned that the random number generator can be seeded to produce a certain sequence over and over. While this is useful in some gaming situations (replaying of card hands, for example) it was totally in the way for my application!

What I wanted was a relatively random selection from my mystery list of words, even upon cold power on situations! If each time the player booted-up, he/she faced the same sequence, then the game took on a truly boring character. Obviously the random generator is seeded with some value upon booting, and I could always place whatever value I wanted in the program to reseed the generator — but how would I get a random seed value out of a cold methodical Apple???

It seems that most of the literature on the RND function concentrates on the problem of degeneration — i.e. that after prolonged usage, the RND function will tend to get less and less random producing a pattern of numbers. This has been discussed in David Lingwood's Amplifying Applesoft *(Call -A.P.P.L.E. in Depth #1),* and also just recently in the May issue of *Call -A.P.P.L.E.* (in David Sparks' article: Random Adventures). His discussion of seeding is, I believe, misleading. He stated that "Seeding Basic's function is simply a matter of POKEing locations $4E and $4F."* This is not exactly correct. It is *not* what he did himself in his BASIC program examples (page 49). He seeded the RND function by PEEKing these locations and using that value as a negative argument to seed the generator sequence! In fact the generator can be seeded by any negative argument as explained in the Basic Programming Reference Manual for Applesoft (page 159).

I would refer fellow readers to a past issue of *Call -A.P.P.L.E.* (February 1981, page 24) where Tom Crawford came up with a clever solution for the non-random RND function. He refers to these same locations ($4E and $4F) as a "counter" which is always being updated whenever the Apple is awaiting keyboard input. PEEKing these locations gives one a value between 0 and 65535. PEEKing only one of these locations will give you a single byte between 0 and 255 which may be enough as we will see.

The idea is simply to use the value in this counter to determine the end point of a random loop. When the loop is exited, you will be at a relatively random spot in the not so random sequence that the Apple has generated with its initial seed value. The key is to have areas in your program before calling the RND function where the program awaits a user's input, or whatever from the keyboard (even if only to read the title!).

Then you will have some random values at these two locations... Now the rest is easy! Once I understood this, it was simple to apply Crawford's solution to generate a more random sequence for my word list in my game.

I hope other readers who have a similiar need for a random sequence will benefit from the use of this counter value held in locations $4E and $4F. It is an easy way out of the cold boot sequence problem about which the definitive word has I'm sure not yet been heard!!

By the way, I enjoyed experimenting with David Sparks' random test in Applesoft using Hi-Res! I ran these tests on the random function with no seeding, with a seed of -1 and with selected negative seeds. I also tried Dr. Norman Holthouse's routine (from D. Lingwood's article). After about 25000 generations, there was no great difference in the distribution of numbers being selected. I did notice a difference between using the RND function and Holthouse's algorithm. This routine was written in BASIC and hence much slower than the call to the Applesoft RND function. Besides that difference, Holthouse's routine generated a specific pattern spread on the Spark's black hole test after only a few hundred calls. I don't know what to make of the pattern exactly, perhaps some other curious reader might delve into the issue. I haven't tried David Sparks' assembly routines yet as I don't dabble in machine language. In any case my problem was solved by that back issue of *Call -A.P.P.L.E.* with Crawford's suggestion. Thanks Tom, David, and David... for the learning experience — and may you all have *truly Random* adventures!!

Marc Weiner
11642 Kiowa Ave. apt.9
Los Angeles, CA 90049

** Sparks was here referring to Integer Basic, not Applesoft. ...ed*

Dear Val,

When we at *Superior Software Inc.,* first introduced the *The Quest for the Holy Grail* in December of 1981, we decided not to copy-protect our disks!

It is our sincere belief that the microcomputer industry as a whole, and the Apple community in particular, will benefit from the most complete exchange of ideas and information possible, consistent with the legitimate rights of software producers to adequate compensation. As authors and publishers, we recognize and understand the arguments for protection; but as users and enthusiasts, we oppose uncopyable and, by their nature, unlistable and unmodifiable programs.

One of the greatest pleasures of owning an Apple or other personal computer comes from learning, and one of the best ways to learn is to review and try to understand what others have done previously. Then you can customize programs to suit your own specific needs or write your own with the knowledge gained. This is the way most of the early greats in micros learned. However, this is impossible when you can't list the program or make your own modifications. The learning process ends before it begins.

Furthermore, a great many commercial programs would be much more valuable to the user if modifications were possible (not to mention the much discussed problem of backing-up valuable, uncopyable disks). Locking the program in these instances actually makes the program less valuable to the user and may even reduce future sales. We believe this to be undesirable and that the risks involved in "going bare" are outweighed by the value to the user and the basic fairness of providing quality software which can be listed, understood, modified, and legitimately backed-up.

TYPESETTING
from your own
COMPUTER

Type your manuscripts, catalogs, or other text into your word processing computer.

Send it to us on a diskette or via- a modem transmission and we'll send you camera ready film copy.

Computer typesetting provides high quality, easy to read copy at a fraction of the cost of conventional typesetting.

Call or write for instructional brochure that includes typestyle selections, formatting instructions, and examples.

Dream Electronics
503-7524833
131 N.W. 2 nd. St.
Corvallis, Oregon 97330

WRITE -A.P.P.L.E.

We are not, under any circumstances, condoning piracy! Duplicating another's copyrighted work and distributing it (even for free) without permission is, and should be, illegal. Pirates are not romantic heroes; they are thieves who should be punished. However, we believe the vast majority of users are honest and will pay a fair price for good software. We are building our company on that belief. . . and you.

Although almost all major software producers use some form of protection, there have been several recent announcements to the contrary. Apple's Mike Markkula stated recently that he would like to see the elimination of locked software altogether. Also, Mark Pelczarski of Penguin Software announced in March, Penguin's decision to "go unprotected" with certain software. We applaud these efforts. It takes courage and trust in you, the user.

As *Superior Software* prepares to expand its product line with the introduction this summer of *Doom Valley, Useable Graphics, and Asteroid Belt,* we once again pledge to you, our customers, that we will continue to produce our software unlocked. Trust must start somewhere, so we will start by trusting you. This trust is also the reason we will ship orders received via *The Source* and telephone even before your check arrives, and why there is no waiting for your personal check to clear. *Superior Software* will continue to bring you superior products at reasonable prices, together with honest and fair business practices. We trust you will be just as fair and honest with us.

Thorne D. Harris III
President
Superior Software Inc.
4312 Arizona Ave.
Kenner, LA 70062

Call -A.P.P.L.E. encourages software manufacturers to advise us of policy changes in re to unprotecting disks.

Dear Val,

There is a small typo in my "So What Did You Expect" on page 70 of the May Call -A.P.P.L.E.

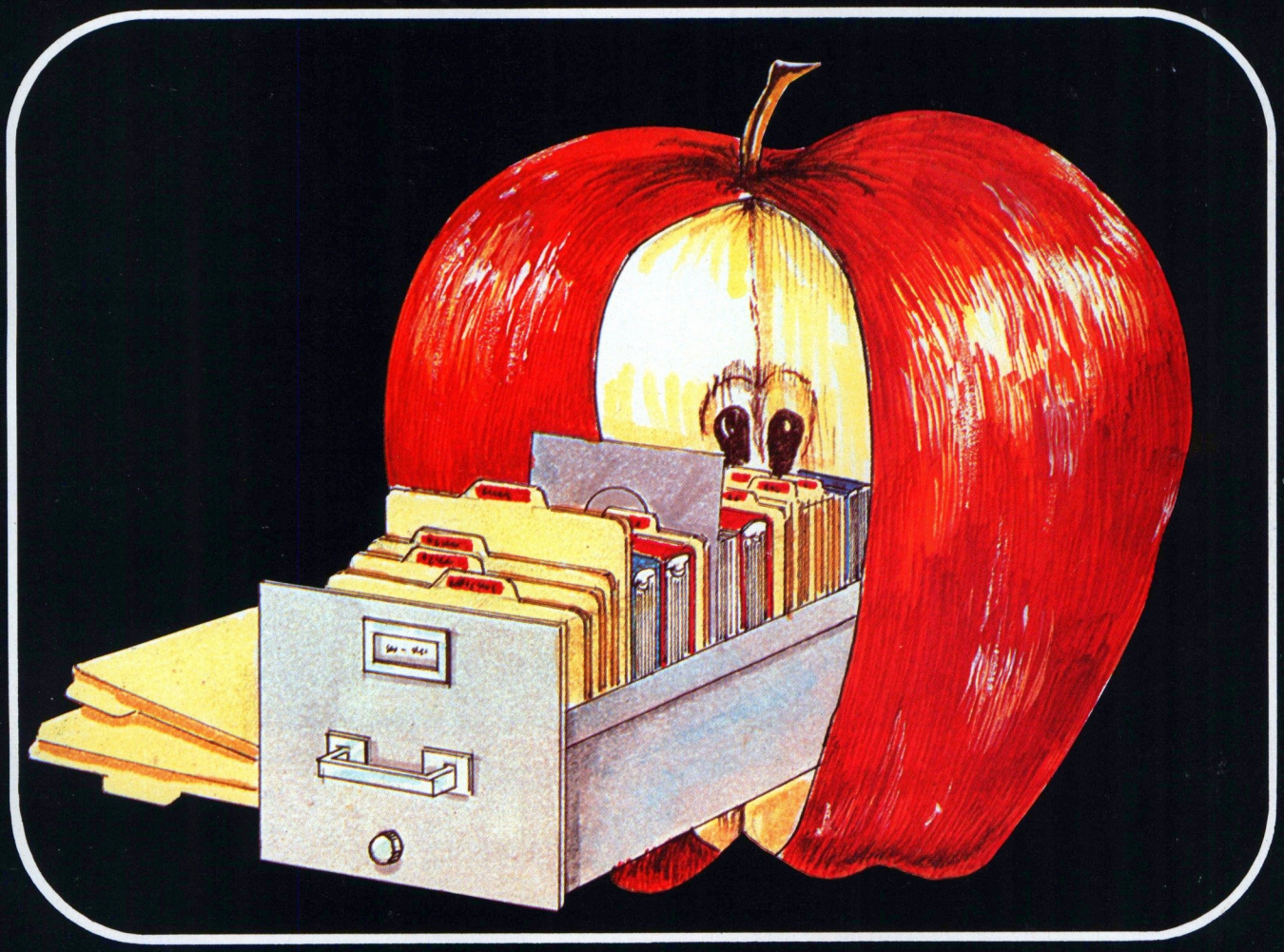
Line 50 should read:

50 FOR K = T TO B STEP 2

There there is no need to add line 55. Hope this helps the readers out there.

Ken Jackson
8955 Valleyview Ave.
Whittier, CA 90605

Now your Apple can give you any record information you need... any way you need it!



The Data Factory™ from Micro Lab adapts your Apple II-48K to your special record keeping needs.

- Exclusive flexible entry positions more information *anywhere* on the monitor.
- Uniquely flexible retrieval; selects records by *any* criteria.
- Tailors reports to *any* format you want. You need no program knowledge.
- Lets you do more with more information, more ways, than any other data base system offered.
- FREE one-year Extended Warranty replaces blown disks and gives you updated versions at no cost to you.

AVAILABLE NOW... Floppy Disk Version $300
Hard Disk Version $500

THE DATA FACTORY™

another fine product from

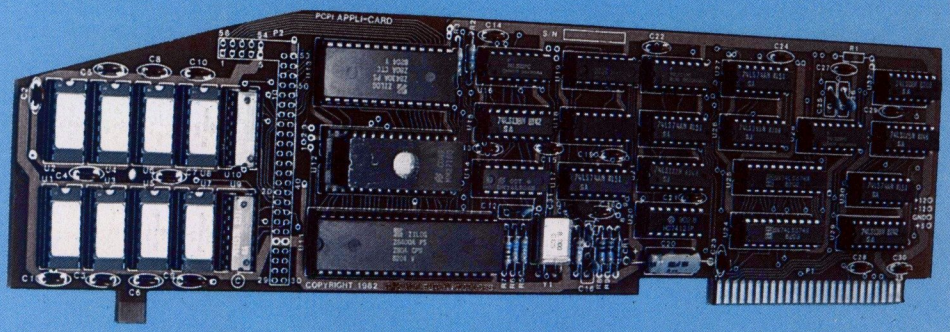
micro lab
systems that work

2310 Skokie Valley Road
Highland Park, IL 60035
(312) 433-7550

NEW . . . For Apple II & III*
from **PERSONAL COMPUTER PRODUCTS, INC.**

APPLI-CARD™

4 or 6 mhz Z-80 and 64K ON-CARD MEMORY



THE ONE CARD SOLUTION TO EXECUTE CP/M* APPLICATION PROGRAMS

- Development Languages Available
- Applications Available
- Compare Our Features And Get The Best Value And Performance For Your Money.

FEATURES	Z-CARD*	SoftCard*	APPLI-CARD
6 mhz Z-80 available	No	No	Yes
64K on-card memory	No	No	Yes
CP/M available	Yes	Yes	Yes
SB-80* with card	No	No	Yes
40 col. to 255 col. horizontal scroll	No	No	Yes
Choice of application	No	No	Yes
2K PROM on the card	No	No	Yes
Real time clock available on the card	No	No	Yes
Expansion interface on the card	No	No	Yes
70 col. upper & lower case	No	No	Yes
A self-contained Z-80A or Z-80B with memory	No	No	Yes
One-card Wordstar* execution	No	No	Yes
63K available for program development or execution	No	No	Yes
Menu driven set up	No	No	Yes

Call today for pricing and product information. Dealer inquiries invited.

PERSONAL COMPUTER PRODUCTS, INC.
16776 Bernardo Center Drive,
San Diego, California 92128
(714) 485-8411 or call your local dealer

*Registered Trade Marks: Apple II & III Apple Computer, Inc. (APPLI-CARD for Apple III will be available fall 1982), CP/M® Digital Research, Inc., Z-CARD Advanced Logic Systems, SoftCard Microsoft Consumer Products, SB-80 Lifeboat Associates, Wordstar MicroPro, Inc.

FUTRA CO.

GONE FISHING SALE!

Who Needs a Baked Apple?
Stay Cool With a **Kensington Fan.**
$74.00 List $89.00

Videx Videoterm $219.95
Electronic Switch $25.00 w/Videoterm purch.
Keyboard Enhancer $109.00 w/Videoterm purch.

Microsoft Z-80 Softcard $269.00
CP/M User's Guide FREE w/Softcard purch.
16K Ramcard $119.00 w/Softcard purch.

New! Commodore 64 $Call

Mail Order
FUTRA COMPANY
P.O. BOX 4380-C
Torrance, CA 90510

Order Line:
(800) 421-5006
(213) 328-8951

Retail
OMC
20695 S. Western #124
Torrance, CA 90505

Retail:
(213) 328-1760

Terms: Shipping add 3% for product shipped within continental U.S.A. (AK & HI excluded) via UPS surface. (minimum $6.00). **If the order placed is prepaid with U.S. funds in form of check or money order, a total charge of $6.00 is all you pay within the continental U.S.A. via UPS surface.** Allow 14 working days for personal and company checks to clear. Credit card charges limited to $1000 maximum. We cannot accept COD orders. FPO, APO or orders outside continental U.S.A. (including AK & HI) call or write for shipping charges or add 20% to purchase price (any difference will be refunded). CA residents add 6% sales tax. All offers subject to change or withdrawal without notice.

AP. COOLKIT

Available July 15th
A.P.P.L.E. Members

$22.50

Consisting of a Torin slow speed fan, line cord and mounting tape. (See Keep Your Cool, Call — A.P.P.L.E., May 1982)

A.P.P.L.E Orders
304 Main Ave. S., Suite 300
Renton, WA 98055
(206) 271-4514

Washington State Residents
add 6.5% sales tax

All foreign orders (including Canada) require additional postage & shipping
Shipping Wt. 16 oz.

AD INDEX

A.P.P.L.E. 16, 41, 76, 78, 87, 88
Advanced Operating Systems 48, 66
Adventure International 57, 81
Amdek Corp. 4
Andromeda 53
Apple Tree Electronics 67
Appleware, Inc. 76
Axlon 61
Beagle Brothers 28
Bit 3 Computer Corp. 68
Businessmaster 74
Bytes & Pieces 45
Cases, Inc. 59
Computer Case Co. 74
Computer Consulting Services 36
Connecticut Information Systems 14
Cortland 62
Datamost ..... IBC, 84
Discount Data Products 32
Dream Electronics 84
Fourth Dimension 1
Futra Company 87
The Geegery Software Works 12
Hayden Book Company Inc. 13
Interface, Inc. 64
Kensington Microware 7
Korsmeyer Electronic Design 64
Krell Software Corp. 63
Leading Edge Products ..... IFC, BC
Link Systems 17
Madwest Software 46
Mark Four Imports 68
Micro Lab 85
Micro Mantic Computer 80
Microware Distributing 76
Omega Microware 50
Orange Micro, Inc. 6
Personal Computer Products 86
R.H. Electronics 65
Sensible Software 18, 44
S & H Software 73
SofTech Microsystems 29, 31, 33, 35, 37
Soft Key Publishing 74
Softronics, Inc. 82
Software & Peripherals 58
Southeastern Software 39
Southwestern Data Systems 25, 56, 66
SSM Microcomputer Corp. 23
Strategic Simulations Inc. 42
Strobe, Inc. 8
Sympathetic Software 78
Telephone Software Connection 49
Vista Computer Company 47
Westware Inc. 61
XPS, Inc. 54
Yucaipa Software 56

A.P.P.L.E. ORDERS
304 Main Ave. S., Suite 300
Renton, WA 98055
(206) 271-4514

I am enclosing my check/MO for $ _____

My membership number is: _____ exp. date _____

Please send me the following: _____

_____

_____

_____

Name: _____

Address: _____

City: _____ State: _____ Zip Code: _____

Phone: _____

MasterCard acc. # _____

VISA® acc. # _____ exp. date _____

Washington state residents must add 6.5% state sales tax.
All foreign orders will require shipping and handling charges (including Canada).

For A.P.P.L.E Members Only

DISKPAK 12 - $16.50

Newly Revised for Apple II+*

A collection of 40 Applesoft programs including games, 8 color demos, etc.

A.P.P.L.E. Orders
304 Main Ave. S., Suite 300
Renton, WA 98055
(206) 271-4514
Visa, M/C welcome
Washington State residents add 6.5% tax.
Overseas shipping wt. 1 oz.
All foreign orders (incl. Canada)
will require additional postage and shipping.

*Apple II+ is registered trademark of Apple Computer Inc.

UNCLASSIFIEDS — A.P.P.L.E. ORDERS

CALL -A.P.P.L.E. (un)Classifieds

Members **Non-commerical** rate .05/word
Members commerical rate .25/word
All others: 1.00/word, $25 minimum
Send ad copy with payment to editorial office

PEACHY WRITER — New! A simple, easy to use text editor at a price you can afford! $24.95. HAPPY FACE — 4 hi-res games for children: Riddles, Famous Sayings, Nursery Rhymes, Spelling. 48K $15.00 COLLEGE PHYSICS — 11 disks, 75 programs, extensive graphics $200. AQUARIUM 25; DINOSAURS $15; CHRISTIAN EDUCATION $15. Write for a free catalog. VISA M/C. Cross Educational Software, Box 1536, Ruston, LA 71270 (318) 255-8921.

WANTED: Volunteer consultant to specialize in Data Base questions. Contact Val Golding (206) 932-6588.

"PUT AN ATLAS IN YOUR APPLE! Now you can quickly locate North American cities and determine exact distances between them. Includes a 411 city Data Base and Hi-Res MAP. Ten day moneyback GUARANTEE! ONLY $24.95 (check/VISA/MC). DOS 3.2/3.3. For FREE information/ order write: Micro-ATLAS, Columbia Software, Box 2235E, Columbia, Maryland 21045. (301) 997-3100"

SEE THE CORE OF YOUR APPLE! Just released: A 400 square inch poster of the 6502 CPU's micro topology. See the Apple's microprocessor enlarged over 20,000 times in full color. You've played with it, programmed it, and cursed at it. Now discover what the little beastie looks like. Search for that infamous "bit bucket" or try to find out where all those software bugs hide. Each order includes a free fact sheet that explains chip history, component location and topology. Satisfaction guaranteed, poster will arrive undamaged, one week delivery. Send $4.50 to: Omicron Digital Systems ltd., P.O. Box 3055, Falls Church, VA 22043.

APPLE ITEMS: Graphics tablet — $490.00; Applesoft ROM card — 110.00; Integer ROM card (never used) — 120.00; Centronics card — 120.00; CCS Analog/Digital card — 90.00; Mt. Hardware ROMPLUS (never used) — 115.00; Desktop Plan (never opened) — 75.00; Strategic Simulation games — $11-$19; IDS 440 w/graphics — $590.00. 513-932-0345 after 7:00 PM.

SILENTYPE excellent condition, includes manual, etc. $175. D.W. Harsin, 5325 Oakhurst Dr., Cambria, CA 93428 (805) 927-3953.

PRINTER — Apple Silentype with interface, manual and 6 rolls (80') paper. $300 or best offer. Tom Little, 800 Cielo Circle, Las Cruces NM 88005. (505) 523-0602.

MICRO/MAILER + is a complete mailing list system. Functions: create, delete, catalog lists; add, correct, delete, sort, display, and list records. Prints labels 1, 2, or 3 up; stores 900 records per disk. Even interfaces with Magic Window files to create customized form letters! $29.95 — Prairie MicroSystems, Inc., 8711 Black Maple Drive, MN 55344.

APPLE GRAPHICS TABLET — Complete with Apple and On-Line Systems software. $500. (206) 941-1084.

APU/LTYP — Smart typewriter program. Lower case printing, capitals inverse on display. Disk save, load, combine, backspace, tabs, line insert, delete, shift lock. Wrap around for heads down typing. $19.95 on 16 sector diskette with copyable source code and documentation. 48K Applesoft, disk printer. K M Systems, 7955 E. 50th St., Suite F-1006, Tulsa, OK 74145.

GENERAL LEDGER SYSTEM — menu driven, error checking on data entry, user formatted statements. Developed by CPA and CDP. Copyable source code furnished on 16 sector diskette. 48K Applesoft, one disk drive. $39.95 or documentation only $10.00 refundable on purchase. K M Systems, 7955 E. 50th St., Suite F-1006, Tulsa, OK 74145.

FOR RENT: APPLE II PLUS computers: by the day, week or month. Expert Systems, P.O. Box 345, Redmond, WA 98052, (206) 382-6616.

250 APPLE CASSETTE PROGRAMS, every kind imaginable. First $200 takes all. Jerry Strong, 4229 Baker Ave. N.W., Seattle, WA 98107 (206) 784-2081.

STOLEN: One Apple computer and related equipment taken from room 407 Physics on 4/17/82. Equipment includes: Apple II+ (48K), serial no. A2S2-257308, Disk II with controller for above, ser. 442872, second disk unit, serial 437984, Zenith hi-res monitor (green phosphor), ser. 3616598, Lexicon LEX-11 telephone coupler. The Apple contained a Language System Card, a Tymac parallel printer interface card, and a hand-built interface card. A $100 reward is offered for recovery of this equipment. Contact: R.J. Wilkes, Physics Dept., FM-15, University of Washington, 543-4963.

NEEDED: Apple Computer System. Looking for someone who wants to sell their Apple, but who can afford to let me pay payments for the first six months or so and then pay it off. I have several original hardware enhancements and software ideas that I need to develop, but can't because I don't have my own system. Think of it as sort of an investment. Will pay over retail price for your inconvenience. Terms negotiable. Also—looking for computer technician and/or digital electronics engineer. Degrees not essential, but you MUST know your stuff. Familiarity with Apple computer and video/ graphics systems preferable. Should be located in the Chico, CA area. Write to: Bill Donnelly, POB 1978, Oroville, CA 95965 or call (916) 534-9254.

MICROSOFT Z-80 card for sale. All software and manuals included. Perfect condition. $260. Write: Karl Merris, 2384 Opalo Way, San Diego, CA 92111. Phone: (714) 462-5400 (days).

WANTED: MAGAZINE WRITERS for Technical Analysis of Stocks and Commodities. Any subject matter related to application and interpretation of chart and numerical analysis of stock or commodity data. Write to: P.O. Box 46518, Seattle, WA 98146-0518.

REQUEST FOR CHANGE OF ADDRESS

Dear A.P.P.L.E.,
I am moving and my present address is:

PRESENT:

NAME _____

ADDRESS _____

CITY _____

STATE _____ ZIP CODE _____

EFFECTIVE DATE _____

NEW ADDRESS:

NAME _____

ADDRESS _____

CITY _____

STATE _____ ZIP CODE _____

MEMBERSHIP NUMBER _____

*Please fill out and return this notice two weeks before you move.

PRODUCTS NOW AVAILABLE TO A.P.P.L.E. MEMBERS

Call —A.P.P.L.E. Specials

Wozpak Closeout	$20.00
A.P.P.L.E. Special[1]	$52.50

A.P.P.L.E. LIBRARY

A.P. Drive Technical Manual	$30.00
"Apple Orchard" No. 1 (magazine)	$2.50
Apple Users Guide	$13.00
Assembly Lines: The Book	$16.50
Beneath the Apple DOS	$17.00
"Best of Cider Press"	
(S.F. Newsletter Annual) Vol I	$4.50
(S.F. Newsletter Annual) Vol II	$9.50
Call —A.P.P.L.E. annuals	
1978	$7.50
1979	$10.00
1980	$15.00
1981	$18.50
Call —A.P.P.L.E. in Depth No. 1	$7.00
Inside Washington Apple Pi	$7.00
Power of VisiCalc Vol I or II, each	$8.50
Using 6502 Assembly Language	$17.00
What's Where in the Apple	$12.50
Woz Talks (VHS Video cassette)	$27.50

AS IS SOFTWARE

Public domain diskettes (DOS 3.2.1) (Send 37¢ S.A.S.E. for list of titles)	ea. $4.00

Diskettes supplied on DOS 3.3 except as noted.

A.P.P.L.E. DISKPAKS

Call —A.P.P.L.E. Anthology Vols I-VII, incl.	ea. $12.50
DICE 82[2]	$12.50
Diskpak 12	$16.50
Diskpak 34A	$13.00
Diskpak 5	$13.00
Diskpak 6B (DOS 3.2 disk utility)	$18.50
Diskpak 6B-16 (DOS 3.3 disk utility)	$18.50
Diskpak 7	$17.50
Diskpak 8	$13.00
In Depth 1 Diskette	$16.50

A.P.P.L.E. HARDWARE

ApController	$95.00
Ap.Coolkit	$22.50
AP.DRIVE	$395.00
AP.DRIVE with Controller	$475.00
Ap L Case 1	$30.00
Ap L Case 2 (Lower case chip)	$27.50
Disk Banks	2 for $15.00

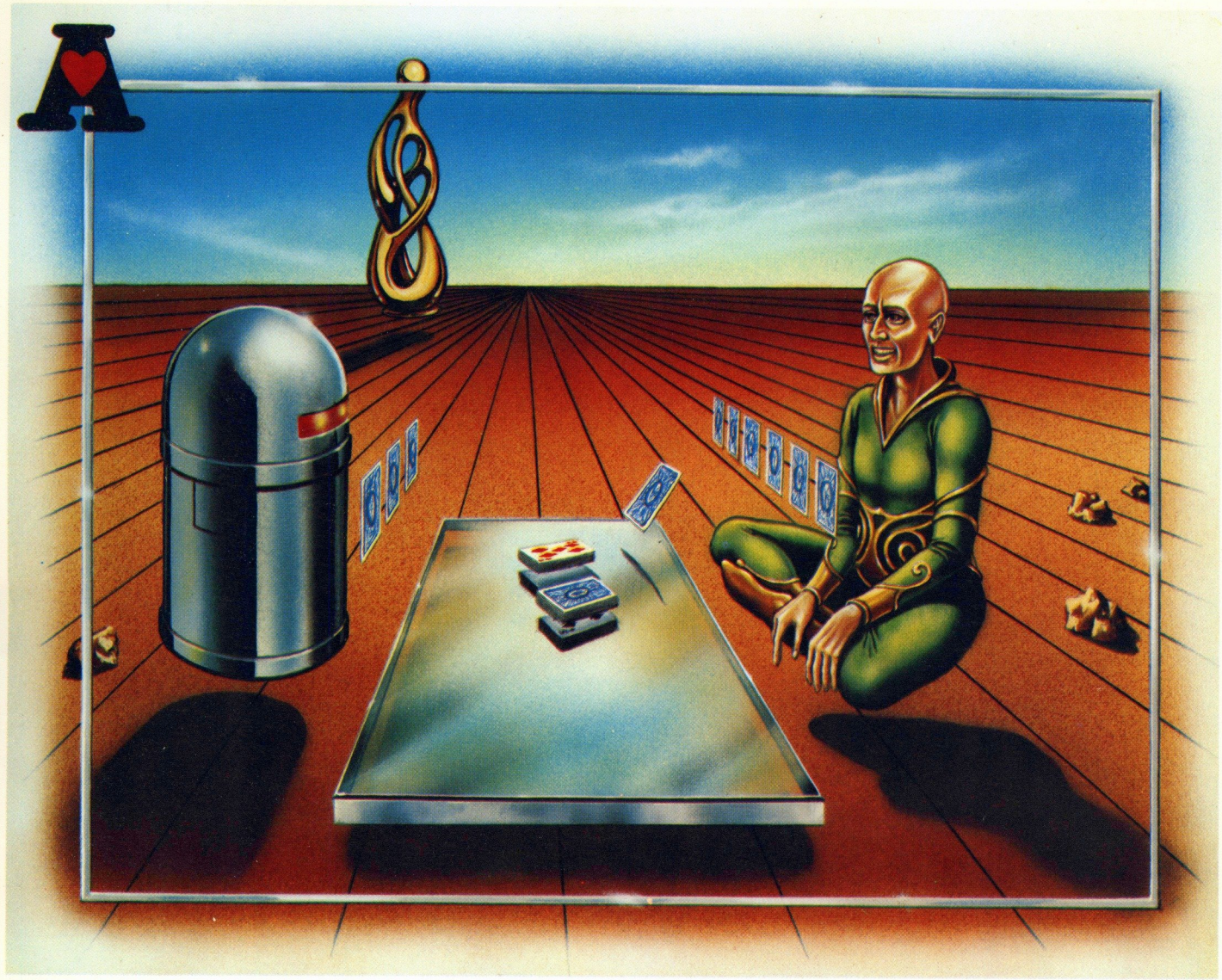
Apple Prestige Product LinE

APfile — Text file utility	$18.50
Applesoft Carpenter	$18.50
Aptest	$18.50
Big Mac Macro-Assembler/Ted	$22.50
Big Mac.LC	$28.50
Big Mac X-Ref	$7.50
Global Program Line Editor	$38.50
Higher Fonts 1	$13.00
Higher Text Plus	$27.50
Program Global Editor	$18.50
Program Line Editor	$20.00
Soft Seventy	$22.00
Symbol Symon	$20.00
WordWeaver ///	$75.00

[1]Must be postmarked or phone ordered by 8/31/82
[2]$7.50 With Ap L Case 1
$10.00 With Ap L Case 2

To avoid shipping delays, always order new products separately.

ORDER FORM ON REVERSE



List off your favorite card games and you'll agree not one has the fun and enjoyment offered by Gin Rummy . . . provided you're up against a tough opponent.

Good Gin players have always been agonizingly hard to find, great ones almost impossible . . . especially at the times you're in the mood. Until Now. **Now** you have **Computer Gin Rummy!** – the perfect opponent who's ready to play whenever you are, for as long as you want!

Computer Gin Rummy is no simple "Go Fish" type of competitor . . . but a master tactician who plays like a pro, who has all the strategy, knows all the moves to give you a terrific battle every time! He knocks when you least expect it, holds out to undercut you – he discards the right cards (which are wrong for you!), and suddenly lays down his hand, as if to say, "name of the game," and gins! This is real Gin Rummy where you can change your lay-offs, rearrange your cards at will, even take back a bad discard before it's fully in play.

Once you start playing **Computer Gin Rummy** you'll find all other card games seem dull and boring, because none give you such a smart competitor. Now for some extra good news. Along with GIN RUMMY, you also get KNOCK RUMMY and ONE-MELD RUMMY so you can relax with a change of pace.

Get your copy now and let's DEAL! Only $29.95 for the apple II,* at your computer store, or from:

DATAMOST INC

9748 Cozycroft Avenue
Chatsworth, California 91311
(213) 709-1202

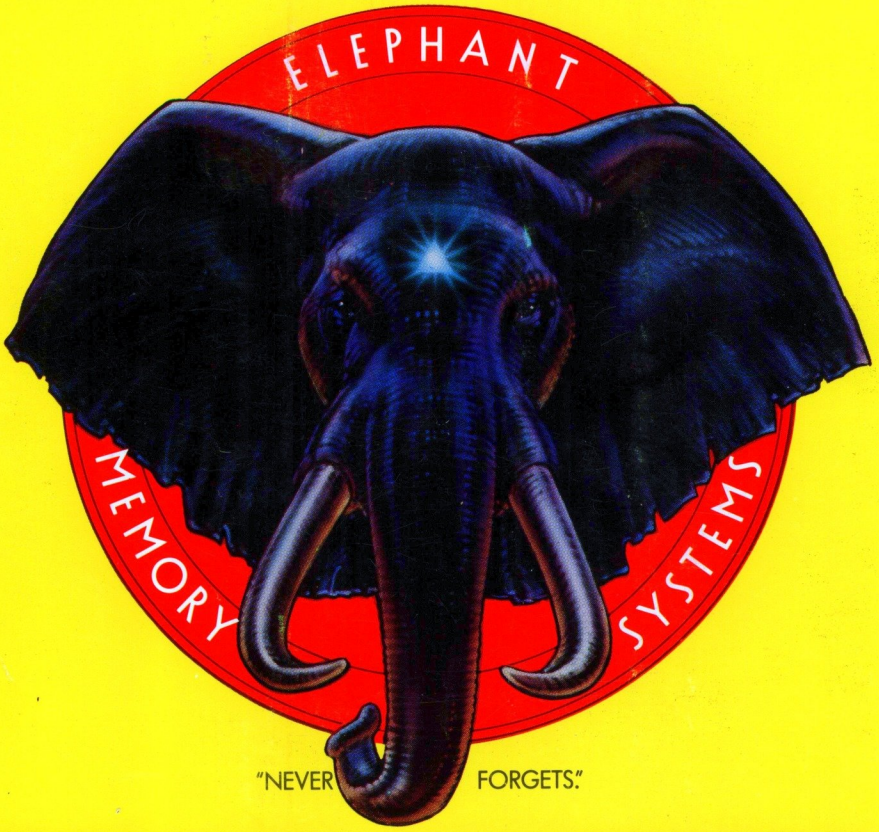
VISA/MASTERCHARGE accepted. $1.00 shipping/handling charge. (California residents add 6% sales tax.)

*Apple II as a trademark of Apple Computer, Inc.

Makes all other card games seem boring . . .

COMPUTER GIN RUMMY

REMEMBER:



"NEVER FORGETS."

MORE THAN JUST ANOTHER PRETTY FACE.

Says who? Says ANSI.

Specifically, subcommittee X3B8 of the American National Standards Institute (ANSI) says so. The fact is all Elephant™ floppies meet or exceed the specs required to meet or exceed all their standards.

But just who is "subcommittee X3B8" to issue such pronouncements?

They're a group of people representing a large, well-balanced cross section of disciplines—from academia, government agencies, and the computer industry. People from places like IBM, Hewlett-Packard, 3M, Lawrence Livermore Labs, The U.S. Department of Defense, Honeywell and The Association of Computer Programmers and Analysts. In short, it's a bunch of high-caliber nitpickers whose mission, it seems, in order to make better disks for consumers, is also to make life miserable for everyone in the disk-making business.

How? By gathering together periodically (often, one suspects, under the full moon) to concoct more and more rules to increase the quality of flexible disks. Their most recent rule book runs over 20 single-spaced pages—listing, and insisting upon—hundreds upon hundreds of standards a disk must meet in order to be blessed by ANSI. (And thereby be taken seriously by people who take disks seriously.)

In fact, if you'd like a copy of this formidable document, for free, just let us know and we'll send you one. Because once you know what it takes to make an Elephant for ANSI . . .

We think you'll want us to make some Elephants for you.

ELEPHANT.™ HEAVY DUTY DISKS.

Distributed Exclusively by **Leading Edge Products, Inc.**, 225 Turnpike Street, Canton, Massachusetts 02021
Call: toll-free 1-800-343-6833; or in Massachusetts call collect (617) 828-8150. Telex 951-624.